HUMAN COMPUTER AND INTEGRATION 2

Numeriano B. Aguado
Chrisna L. Fucio

Bachelor of Science in Information Technology
College of Computing Studies

v2.01182024

Table of Contents

MODULE 5
INTERFACE AND DATA GATHERING

Introduction

When considering how to solve a user problem, the default solution that many developers choose to design is an app that can run on a smartphone. Making this easier still are many easy-to-use app developer tools that can be freely downloaded. It is hardly surprising, therefore, to see just how many apps there are in the world. In December 2018, Apple, for example, had a staggering 2 million apps in its store, many of which were games.

Despite the ubiquity of the smartphone app industry, the web continues to proliferate in offering services, content, resources, and information. A central concern is how to design them to be interoperable across different devices and browsers, which takes into account the varying form factors, size, and shape of smart watches, smartphones, laptops, smart TVs, and computer screens. Besides the app and the web, many other kinds of interfaces have been developed, including voice interfaces, touch interfaces, gesture interfaces, and multimodal interfaces.

The proliferation of technological developments has encouraged different ways of thinking about interaction design and UX. For example, input can be via mice, touchpads, pens, remote controllers, joysticks, RFID readers, gestures, and even brain-computer interaction. Output is equally diverse, appearing in the form of graphical interfaces, speech, mixed realities, augmented realities, tangible interfaces, wearable computing, and more.

Learning Outcomes

At the end of this module, students should be able to:

1. Provide an overview of the many different kinds of interfaces.
2. Highlight the main design and research considerations for each of the interfaces.
3. Discuss what is meant by a natural user interface (NUI).
4. Consider which interface is best for a given application or activity.

Lesson 1. Interface Types (Sharp, 2019)

Numerous adjectives have been used to describe the different types of interfaces that have been developed, including *graphical*, *command*, *speech*, *multimodal*, *invisible*, *ambient*, *affective*, *mobile*, *intelligent*, *adaptive*, *smart*, *tangible*, *touchless*, and *natural*. Some of the inter- face types are primarily concerned with a function (for example, to be intelligent, to be adaptive, to be ambient, or to be smart), while others focus on the interaction style used (such as command, graphical, or multimedia), the input/output device used (for instance, pen-based, speech-based, or gesture-based), or the platform being designed for (for example, tablet, mobile, PC, or wearable). Rather than cover every possible type that has been developed or described, we have chosen to select the main types of interfaces that have emerged over the past 40 years. The interface types are loosely ordered in terms of when they were developed. They are numbered to make it easier to find a particular one. (See the following list for the complete set.) It should be noted, however, that this classification is for convenience of reference. The interface entries are not mutually exclusive since some products can appear in two or more categories. For example, a smartphone can be considered to be mobile, touch, or wearable.

The types of interfaces covered in this chapter include the following:

1. Command
2. Graphical
3. Multimedia
4. Virtual reality
5. Web
6. Touch
7. Haptic
8. Shareable
9. Augmented reality
10. Robots and drones
11. Mobile
12. Appliance
13. Voice
14. Pen
15. Gesture
16. Multimodal
17. Tangible
18. Wearables
19. Smart
20. Brain-computer interaction

- **Command-Line Interfaces**

Early interfaces required the user to type in commands that were typically abbreviations (for example, ls) at the prompt symbol appearing on the computer display, to which the system responded (for example, by listing current files). Another way of issuing commands is by pressing certain combinations of keys (such as Shift+Alt+Ctrl). Some commands are also a fixed part of the keyboard, such as delete, enter, and undo, while other function keys can be programmed by the user as specific commands (for instance, F11 commanding print action). Command-line interfaces were largely superseded by graphical interfaces that incorporated commands such as menus, icons, keyboard shortcuts, and pop-up/predictable text commands as part of an application. Where command-line interfaces continue to have an advantage is when users find them easier and faster to use than equivalent menu-based systems (Raskin, 2000). Users also prefer command-line interfaces for performing certain operations as part of a complex software package, such as for CAD environments, to allow expert designers to interact rapidly and precisely with the software. They also provide scripting for batch operations, and they are being increasingly used on the web, where the search bar acts as a general-purpose command-line facility, for example, www.yubnub.org.

System administrators, programmers, and power users often find that it is much more efficient and quicker to use command languages such as Microsoft's PowerShell. For example, it is much easier to delete 10,000 files in one go by using one command rather than scrolling through that number of files and highlighting those that need to be deleted. Command languages have also been developed for visually impaired people to allow them to interact in virtual worlds.

- **Graphical User Interfaces**

As discussed in "Conceptualizing Interaction", the birth of the graphical user interface (GUI), opening up new possibilities for users to interact with a system and for information to be presented and represented within a graphical interface. Specifically, new ways of visually designing the interface became possible, which included the use of color, typography, and imagery (Mullet and Sano, 1995). The original GUI was called a WIMP (windows, icons, menus, pointer) and consisted of the following:

- *Windows*: Sections of the screen that can be scrolled, stretched, overlapped, opened, closed, and moved using a mouse
- *Icons*: Pictograms that represent applications, objects, commands, and tools that are opened or activated when clicked on
- *Menus*: Lists of options that can be scrolled through and selected in the way a menu is used in a restaurant
- *Pointing device*: A mouse controlling the cursor as a point of entry to the windows, menus, and icons on the screen

The first generation of WIMP interfaces were primarily boxy in design; user interaction took place through a combination of windows, scroll bars, checkboxes, panels, palettes, and dialog boxes that appeared on the screen in various forms (see Figure 4.1). Developers were largely constrained by the set of widgets available to them, of which the dialog box was most prominent. (A widget is a standardized display representation of a control, like a button or scroll bar, that can be manipulated by the user). Nowadays, GUIs have been adapted for mobile and touchscreens. Instead of using a mouse and keyboard as input, the default action for most users is to swipe and touch using a single finger when browsing and interacting with digital content.

Figure 5.1 The boxy look of the first generation of GUIs

The basic building blocks of the WIMP are still part of the modern GUI used as part of a display, but they have evolved into a number of different forms and types. For example, there are now many different types of icons and menus, including audio icons and audio menus, 3D animated icons, and even tiny icon-based menus that can fit onto a smartwatch screen (see Figure 4.2). Windows have also greatly expanded in terms of how they are used and what they are used for; for example, a variety of dialog boxes, interactive forms, and feedback/error message boxes have become pervasive. In addition, a number of graphical elements that were not part of the WIMP interface have been incorporated into the GUI. These include toolbars and docks (a row or column of available applications and icons of other objects such as open files) and rollovers (where text labels appear next to an icon or part of the screen as the cursor is rolled over it). Here, we give an overview of the design considerations concerning the basic building blocks of the WIMP/GUI: windows, menus, and icons.

Figure 5.2 Simple smartwatch menus with one, two, or three options
Source: https://developer.apple.com/design/human-interface-guidelines/watchos/interface-elements/menus/

- ***Window Design***

Windows were invented to overcome the physical constraints of a computer display, enabling more information to be viewed and tasks to be performed on the same screen. Multi- ple windows can be opened at any one time, for example, web browsers, word processing documents, photos, and slideshows, enabling the user to switch between them when needing to look at or work on different documents, files, and apps. They can also enable multiple instances of one app to be opened, such as when using a web browser.

Scrolling bars within windows also enable more information to be viewed than is possible on one screen. Scroll bars can be placed vertically and horizontally in windows to enable upward, downward, and sideway movements through a document and can be controlled using a touchpad, mouse, or arrow keys. Touch interfaces enable users to scroll content simply by swiping the screen to the left or right or up or down.

One of the problems of having multiple windows open is that it can be difficult to find specific ones. Various techniques have been developed to help users locate a particular window, a common one being to provide a list as part of an app menu. macOS also provides a function that shrinks all windows that are open for a given application so that they can be seen side by side on one screen. The user needs only to press one function key and then move the cursor over each one to see what they are called in addition to a visual preview. This technique enables users to see at a glance what they have in their workspace, and it also allows them easily to select one to bring forward. Another option is to display all of the windows open for a particular application, for example, Microsoft Word. Web browsers, like Firefox, also show thumbnails of the top sites visited and a selection of sites that you have saved or visited, which are called *highlights* (see Figure 4.3).

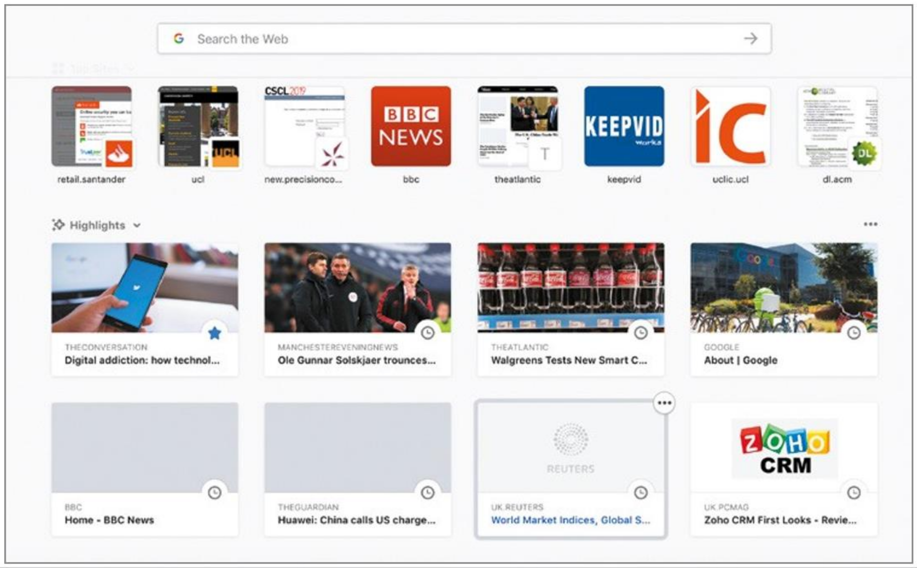


Figure 5.3 Part of the home page for the Firefox browser showing thumbnails of top sites visited and suggested highlight pages (bottom rows)

A particular kind of window that is commonly used is the *dialog box*. Confirmations, error messages, checklists, and forms are presented through dialog boxes. Information in the dialog boxes is often designed to guide user interaction, with the user following the sequence of options provided. Examples include a sequenced series of forms (such as Wizards) presenting the necessary and optional choices that need to be filled in when choosing a PowerPoint presentation or an Excel spreadsheet. The downside of this style of interaction is that there is a tendency to cram too much information or data entry fields into one box, making the interface confusing, crowded, and difficult to read (Mullet and Sano, 1995).

- **Menu Design**

Interface menus are typically ordered across the top row or down the side of a screen using category headers as part of a menu bar. The contents of the menus are also for the large part invisible, only dropping down when the header is selected or rolled over with a mouse. The various options under each menu are typically ordered from top to bottom in terms of most frequently used options and grouped in terms of their similarity with one another; for example, all formatting commands are placed together.

There are numerous menu interface styles, including flat lists, drop-down, pop-up, contextual, collapsible, mega, and expanding ones, such as cascading menus. *Flat menus* are good at displaying a small number of options at the same time or where the size of the display is small, for example on smartphones, cameras, and smartwatches. However, they often have to nest the lists of options within each, requiring several steps to be taken by a user to get to the list with the desired option. Once deep down in a nested menu, the user then has to take the same number of steps to get back to the top of the menu. Moving through previous screens can be tedious.

Expanding menus enable more options to be shown on a single screen than is possible with a single flat menu list. This makes navigation more flexible, allowing for the selection of options to be done in the same window. An example is the *cascading menu*, which provides secondary and even tertiary menus to appear alongside the primary active drop-down menu, enabling further related options to be selected, such as when selecting track changes from the tools menu leads to a secondary menu of three options by which to track changes in a Word document. The downside of using expanding menus, however, is that they require precise control. Users can often end up making errors, namely, overshooting or selecting the wrong options. In particular, cascading menus require users to move their cursor over the menu item, while holding the mouse or touchpad down, and then to move their cursor over to the next menu list when the cascading menu appears and select the next desired option. This can result in the user under or overshooting a menu option, or sometimes accidentally closing the entire menu.

Collapsible menus provide an alternative approach to expanding menus in that they allow further options to be made visible by selecting a header. The headings appear adjacent to each other, providing the user with an overview of the content available (see Figure 4.4). This reduces the amount of scrolling needed. *Contextual menus* provide access to often-used commands associated with a particular item, for example, an icon. They provide appropriate commands that make sense in the context of a current task. They appear when the user presses the Control key while clicking an interface element. For example, clicking a photo on a website together with holding down the Ctrl key results in a small set of relevant menu options appearing in an overlapping window, such as open it in a new window, save it, or copy it. The advantage of contextual menus is that they provide a limited number of options associated with an interface element, overcoming some of the navigation problems associated with cascading and expanding menus.

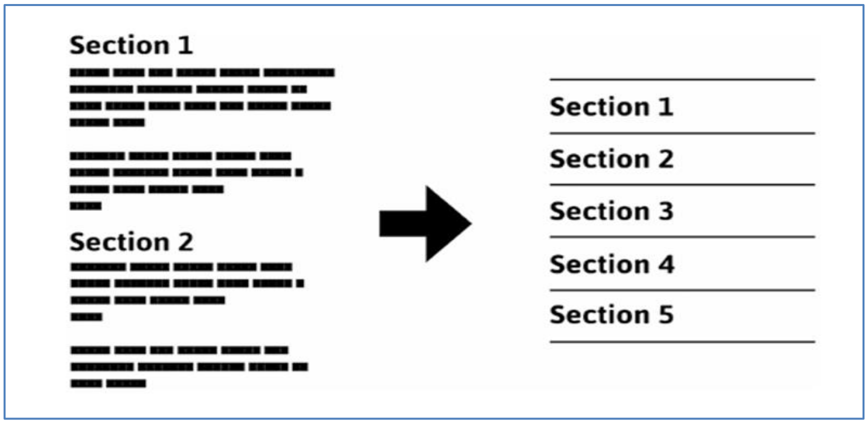


Figure 5.4 A template for a collapsible menu

Source: https://inclusive-components.design/collapsible-sections/. Reproduced with permission of Smashing Magazine

- **Icon Design**

The appearance of icons in an interface came about following the Xerox Star project. They were used to represent objects as part of the desktop metaphor, namely, folders, documents, trashcans, inboxes, and outboxes. The assumption behind using icons instead of text labels is that they are easier to learn and remember, especially for non-expert computer users. They can also be designed to be compact and variably positioned on a screen.

Icons have become a pervasive feature of the interface. They now populate every app and operating system and are used for all manner of functions besides representing desktop objects. These include depicting tools (for example, Paint 3D), status (such as, Wi-Fi strength), categories of apps (for instance, health or personal finance), and a diversity of abstract operations (including cut, paste, next, accept, and change). They have also gone through many changes in their look and feel—black and white, color, shadowing, photorealistic images, 3D rendering, and animation have all been used.

Whereas early icon designers were constrained by the graphical display technology of the day, current interface developers have much more flexibility. For example, the use of anti-aliasing techniques enables curves and non-rectilinear lines to be drawn, enabling more photo-illustrative styles to be developed (*anti-aliasing* means adding pixels around a jagged border of an object to smooth its outline visually). App icons are often designed to be both visually attractive and informative. The goal is to make them inviting, emotionally appealing, memorable, and distinctive.

Different graphical genres have been used to group and identify different categories of icons. Figure 4.5 shows how colorful photorealistic images were used in the original Apple Aqua set, each slanting slightly to the left, for the category of *user* applications (such as email) whereas monochrome straight on and simple images were used for the class of *utility* applications (for instance, printer setup). The former has a fun feel to them, whereas the latter has a more serious look about them. While a number of other styles have since been developed, the use of slanting versus straight facing icons to signify different icon categories is still in use.

Icons can be designed to represent objects and operations in the interface using concrete objects and/or abstract symbols. The mapping between the icon and underlying object or operation to which it refers can be similar (such as a picture of a file to represent the object file), analogical (for instance, a picture of a pair of scissors to represent cut), or arbitrary (for example, the use of an X to represent delete). The most effective icons are generally those that are isomorphic since they have a direct mapping between what is being represented and how it is represented. Many operations in an interface, however, are of actions to be performed on objects, making it more difficult to represent them using direct mapping. Instead, an effective technique is to use a combination of objects and symbols that capture the salient part of an action by using analogy, association, or convention (Rogers, 1989). For example, using a picture of a pair of scissors to represent cut in a word-processing application provides a sufficient clue as long as the user understands the convention of cut for deleting text.

Figure 5.5 Two styles of Apple icons used to represent different kinds of functions

Icons can be designed to represent objects and operations in the interface using concrete objects and/or abstract symbols. The mapping between the icon and underlying object or operation to which it refers can be similar (such as a picture of a file to represent the object file), analogical (for instance, a picture of a pair of scissors to represent cut), or arbitrary (for example, the use of an X to represent delete). The most effective icons are generally those that are isomorphic since they have a direct mapping between what is being represented and how it is represented. Many operations in an interface, however, are of actions to be performed on objects, making it more difficult to represent them using direct mapping. Instead, an effective technique is to use a combination of objects and symbols that capture the salient part of an action by using analogy, association, or convention (Rogers, 1989). For example, using a picture of a pair of scissors to represent cut in a word-processing application provides a sufficient clue as long as the user understands the convention of cut for deleting text.

Another approach that many smart phone designers use is flat 2D icons. These are simple and use strong colors and pictograms or symbols. The effect is to make them easily recognizable and distinctive. Examples shown in Figure 4.6a include the white ghost on a yellow background (Snapchat), a white line bubble with a solid white phone handset in a speech bubble on a lime-green background (WhatsApp), and the sun next to a cloud (weather).

Figure 5.6 2D icons designed for (a) a smartphone and (b) a smartwatch
Source: (a) Helen Sharp (b) https://support.apple.com/en-ca/HT205550

Icons that appear on toolbars or palettes as part of an application or presented on small device displays (such as digital cameras or smartwatches) have much less screen real estate available. Because of this, they have been designed to be simple, emphasizing the outline form of an object or symbol and using only grayscale or one or two colors (see Figure 4.6b). They tend to convey the status, tool, or action using a concrete object (for example, the airplane symbol signaling whether the airplane mode is on or off) and abstract symbols (such as three waves that light up from none to all to convey the strength/power of the area's Wi-Fi).

- **Multimedia** (Sharp, 2019).

Multimedia, as the name implies, combines different media within a single interface, namely, graphics, text, video, sound, and animation, and links them together with various forms of interactivity. Users can click links in an image or text that triggers another media such as an animation or a video. From there they can return to where they were previously or jump to another media source. The assumption is that a combination of media and interactivity can provide better ways of presenting information than can a single media, for example, just text or video alone. The added value of multimedia is that it can be easier for learning, better for understanding, more engaging, and more pleasant (Scaife and Rogers, 1996).

Another distinctive feature of multimedia is its ability to facilitate rapid access to multiple representations of information. Many multimedia encyclopedias and digital libraries have been designed based on this multiplicity principle, providing an assortment of audio and visual materials on a given topic. For example, when looking to find information about the heart, a typical multimedia-based encyclopedia will provide the following:

- One or two video clips of a real live heart pumping and possibly a heart transplant operation
- Audio recordings of the heart beating and perhaps an eminent physician talking about the cause of heart disease
- Static diagrams and animations of the circulatory system, sometimes with narration
- Several columns of hypertext, describing the structure and function of the heart

Hands-on interactive simulations have also been incorporated as part of multimedia learning environments. An early example was the *Cardiac Tutor*, developed to teach students about cardiac resuscitation. It required students to save patients by selecting the correct set of procedures in the correct order from various options displayed on the computer screen. Other kinds of multimedia narratives and games have also been developed to support discovery learning by encouraging children to explore different parts of the display by noticing a hotspot or other kind of link. For example, https://KidsDiscover.com/apps/ has many tablet apps that use a combination of animations, photos, interactive 3D models, and audio to teach kids about science and social studies topics. Using swiping and touching, kids can reveal, scroll through, select audio narration, and watch video tours. Multimedia has largely been developed for training, educational, and entertainment purposes.

- **Virtual Reality**

Virtual reality (VR) has been around since the 1970s when researchers first began developing computer-generated graphical simulations to create "the illusion of participation in a synthetic environment rather than external observation of such an environment" (Gigante, 1993). The goal was to create user experiences that feel virtually real when interacting with an artificial environment. Images are displayed stereoscopically to the users—most commonly through VR headsets—and objects within the field of vision can be interacted with via an input device like a joystick.

The 3D graphics can be projected onto Cave Automatic Virtual Environment (CAVE) floor and wall surfaces, desktops, 3D TV, headsets, or large shared displays, for instance, IMAX screens. One of the main attractions of VR is that it can provide opportunities for new kinds of immersive experiences, enabling users to interact with objects and navigate in 3D space in ways not possible in the physical world or a 2D graphical interface. Besides looking at and navigating through a 360-degree visual landscape, auditory and haptic feedback can be added to make the experience feel even more like the real world. The resulting user experience can be highly engaging; it can feel as if one really is flying around a virtual world.

VR simulations of the world can be constructed to have a higher level of fidelity with the objects they represent compared to other forms of graphical interfaces, for example, multimedia. The illusion afforded by the technology can make virtual objects appear to be very life-like and behave according to the laws of physics. For example, landing and take-off terrains developed for flight simulators can appear to be very realistic. Moreover, it is assumed that learning and training applications can be improved through having a greater fidelity to the represented world.

- **Website Design**

Early websites were largely text-based, providing hyperlinks to different places or pages of text. Much of the design effort was concerned with the information architecture, that is, how best to structure information at the interface level to enable users to navigate and access it easily and quickly. For example, many adapted usability guidelines to make them applicable to website design, focusing on simplicity, feedback, speed, legibility, and ease of use. He also stressed how critical download time was to the success of a website. Simply, users who have to wait too long for a page to appear are likely to move on somewhere else.

Since then, the goal of web design has been to develop sites that are not only usable but also aesthetically pleasing. Getting the graphical design right, therefore, is critical. The use of graphical elements (such as background images, color, bold text, and icons) can make a web- site look distinctive, striking, and pleasurable for the user when they first view it and also to make it readily recognizable on their return. However, there is the danger that designers can get carried away with the appearance at the expense of making it difficult to find something and navigate through it.

Steve Krug (2014) discusses this usability versus attractiveness dilemma in terms of the difference between how designers create websites and how users actually view them. He argues that many web designers create sites as if the user was going to pore over each page, reading the finely crafted text word for word; looking at the use of images, color, icons, and so forth; examining how the various items have been organized on the site; and then con- templating their options before they finally select a link. Users, however, often behave quite differently. They will glance at a new page, scan part of it, and click the first link that catches their interest or looks like it might lead them to what they want.

Web designers now have a number of languages available to design websites, such as Ruby and Python. HTML5 and web development tools, such as JavaScript and CSS, are also used. Libraries, such as React, and open source toolkits, such as Bootstrap, enable developers to get started quickly when prototyping their ideas for a website. WordPress also provides users with an easy-to-use interface and hundreds of free templates to use as a basis when creating their own website. In addition, built-in optimization and responsive, mobile-ready themes are available. Customized web pages are available for smartphone browsers that pro- vide scrolling lists of articles, games, tunes, and so on, rather than hyperlinked pages.

Another interface element that has become an integral part of any website is *breadcrumb navigation*. Breadcrumbs are category labels that appear on a web page that enable users to peruse other pages without losing track of where they have come from. The term comes from the way-finding technique that Hansel used in the Brothers Grimm fairy tale *Hansel and Gretel*. The metaphor conjures up the idea of leaving a path to follow back. Breadcrumbs are also used by search engine optimization tools that match up a user's search terms with relevant web pages using the breadcrumbs. Breadcrumbs also extol usability in a number of ways, including helping users know where they are relative to the rest of the web- site, enabling one-click access to higher site levels, attracting first time visitors to continue to browse a website after having viewed the landing page (Mifsud, 2011). Therefore, using them is good practice for other web applications besides websites.

With the arrival of tablets and smartphones, web designers needed to rethink how to design web browsers and websites for them, as they realized the touchscreen affords a different interaction style than PC/laptops. The standard desktop interface was found to not work as well on a tablet or smartphone. In particular, the typical fonts, buttons, and menu tabs were too small and awkward to select when using a finger. Instead of double-clicking inter- face elements, as users do with a mouse or trackpad, tablet and smartphone screens enable finger tapping. The main methods of navigation are by swiping and pinching. A new style of website emerged that mapped better to this kind of interaction style but also one that the user could interact with easily when using a mouse and trackpad. Responsive websites were developed that change their layout, graphic design, font, and appearance depending on the screen size (smartphone, tablet, or PC) on which it was being displayed.

If you look at the design of many websites, you will see that the front page presents a banner at the top, a short promotional video about the company/product/service,

> Tips on designing websites for tablets versus mobile phones can be found here:
> **https://css-tricks.com/a-couple-of-best-practices-for-tablet-friendly-design/**

arrows to the left or right to indicate where to flick to move through pages, and further details appearing beneath the home page that the user can scroll through. Navigation is largely done by swiping pages horizontally or scrolling up and down.

- **Mobile Devices**

Mobile devices have become pervasive, with people increasingly using them in all aspects of their everyday and working lives—including phones, fitness trackers, and watches. Customized mobile devices are also used by people in a diversity of work settings where they need access to real-time data or information while walking around. For example, they are now commonly used in restaurants to take orders, at car rental agencies to check in car returns, in supermarkets for checking stock, and on the streets for multiplayer gaming.

Larger-sized tablets are also used in mobile settings. For example, many airlines provide their flight attendants with one so that they can use their customized flight apps while air- borne and at airports; sales and marketing professionals also use them to demonstrate their products or to collect public opinions. Tablets and smartphones are also commonly used in classrooms that can be stored in special "tabcabbies" provided by schools for safe keeping and recharging.

Smartphones can also be used to download contextual information by scanning barcodes in the physical world. Consumers can instantly download product information by scanning barcodes using their phone when walking around a supermarket, including allergens, such as nuts, gluten, and dairy. For example, the ***GoodGuide*** app enables shoppers to scan products in a store by taking a photo of their barcode to see how they rate for healthiness and impact on the environment. Others include concert tickets and location-based notifications. Another method that provides quick access to relevant information is the use of quick response (QR) codes that store URLs and look like black-and-white checkered squares. They work by people taking a picture using their camera phone that then takes them to a particular website. However, despite their universal appeal to companies as a way of providing additional information or special offers, not many people actually use them in practice. One of the reasons is that they can be slow, tricky, and cumbersome to use *in the actual*. People have to download a QR reader app first, open it, and then try to hold it over the QR code to take a photo, which can take time to open up a webpage.

- **Appliances**

Appliances include machines for everyday use in the home (for example, washing machines, microwave ovens, refrigerators, toasters, bread makers, and smoothie makers). What they have in common is that most people using them will be trying to get something specific done in a short period of time, such as starting a wash, watching a program, buying a ticket, or making a drink. They are unlikely to be interested in spending time exploring the interface or looking through a manual to see how to use the appliance. Many of them now have LED displays that provide multiple functions and feedback about a process (such as temperature, minutes remaining, and so on). Some have begun to be connected to the Internet with companion devices, enabling them to be controlled by remote apps. An example is a coffee maker that can be controlled to come on at a certain time from an app running on a smartphone or controlled by voice.

- **Voice User Interfaces**

A *voice user interface* (VUI) involves a person talking with a spoken language app, such as a search engine, a train timetable, a travel planner, or a phone service. It is commonly used for inquiring about specific information (for instance, flight times or the weather) or issuing a command to a machine (such as asking a smart TV to select an Action movie or asking a smart speaker to play some upbeat music). Hence, VUIs use an interaction type of command or conversation, where users speak and listen to an interface rather than click on, touch, or point to it. Sometimes, the interaction style can involve the user responding where the system is proactive and initiates the conversation, for example, asking the user if they would like to stop watching a movie or listen to the latest breaking news.

The first generation of speech systems earned a reputation for *mishearing* all too often what a person said. However, they are now much more sophisticated and have higher levels of recognition accuracy. Machine learning algorithms have been developed that are continuing to improve their ability to recognize what someone is saying. For speech output, actors are often used to record answers, messages, and prompts, which are much friendlier, more convincing, and more pleasant than the artificially-sounding synthesized speech that was typically used in the early systems.

VUIs have become popular for a range of apps. Speech-to-text systems, such as Dragon, enable people to dictate rather than have to type, whether it is entering data into a spread- sheet, using a search engine, or writing a document. The words spoken appear on the screen. For some people, this mode of interaction is more efficient, especially when they are on the move. Dragon claims on their website that it is three times faster than typing and it is 99 percent accurate. Speech technology is also used by people with visual impairments, including speech recognition word processors, page scanners, web readers, and VUIs for operating home control systems, including lights, TV, stereo, and other home appliances.

A number of speech-based phone apps exist that enable people to use them while mobile, making them more convenient to use than text-based entry. For example, people can voice queries into their phone using Google Voice or Apple Siri rather than entering text manually. Mobile translators allow people to communicate in real time with others who speak a different language by letting a software app on their phone do the talking (for example, Google Translate). People speak in their own language using their phone while the software translates what each person is saying into the language of the other one. Potentially, this means people from all over the world (there are more than 6,000 languages) can talk to one another without having to learn another language.

Voice assistants, like Amazon's Alexa and Google Home, can be instructed by users to entertain in the home by telling jokes, playing music, keeping track of time, and enabling users to play games. Alexa also offers a range of "skills," which are voice-driven capabilities intended to provide a more personalized experience. For example, "Open the Magic Door" is an interactive story skill that allows users to choose their path in a story by selecting different options through the narrative. Another one, "Kids court," allows families to settle arguments in an Alexa-run court while learning about the law. Many of the skills are designed to support multiple users taking part at the same time, offering the potential for families to play together. Social interaction is encouraged by the smart speaker that houses Alexa or Home. Smart speakers sit in a common space for all to use (similar to a toaster or refrigerator). In contrast, handheld devices, such as a smartphone or tablet, support only single use and ownership.

Despite advances in speech recognition, conversational interaction is limited mainly to answering questions and responding to requests. It can be difficult for VUIs to recognize children's speech, which is not as articulate as adults. For example, Druga et al. (2017) found that young children (3–4 years old) experienced difficulty interacting with conversational and chat agents, resulting in them becoming frustrated. Also, voice assistants don't always recognize who is talking in a group, such as a family, and always need to be called by their name each time someone wants to interact with it. There is still a way to go before voice assistant interaction resembles human conversation.

- **Pen-Based Devices**

Pen-based devices enable people to write, draw, select, and move objects on an interface using light pens or styluses that capitalize on the well-honed drawing and writing skills that are developed from childhood. They have been used to interact with tablets and large displays, instead of mouse, touch, or keyboard input, for selecting items and supporting freehand sketching.

Another advantage of digital pens is that they allow users to annotate existing documents, such as spreadsheets, presentations, and diagrams quickly and easily in a similar way to how they would do this when using paper-based versions. This is useful for a team who is working together and communicating to each other from different locations. One problem with using pen-based interactions on small screens, however, is that sometimes it can be difficult to see options on the screen because a user's hand can obscure part of it when writing.

- **Touchscreens**

Single touchscreens, used in walk-up kiosks such as ticket machines or museum guides, ATMs, and cash registers (for instance, restaurants), have been around for a while. They work by detecting the presence and location of a person's touch on the display; options are selected by tapping on the screen. *Multitouch surfaces*, on the other hand, support a much wider range of more dynamic fingertip actions, such as swiping, flicking, pinching, pushing, and tapping. They do this by registering touches at multiple locations using a grid (see Figure 7.19). This multitouch method enables devices, such as smartphones and tabletops, to recognize and respond to more than one touch at the same time. This enables users to use multiple digits to perform a variety of actions, such as zooming in and out of maps, moving photos, selecting letters from a virtual keyboard when writing, and scrolling through lists. Two hands can also be used together to stretch and move objects on a tabletop surface, similar to how both hands are used to stretch an elastic band or scoop together a set of objects.

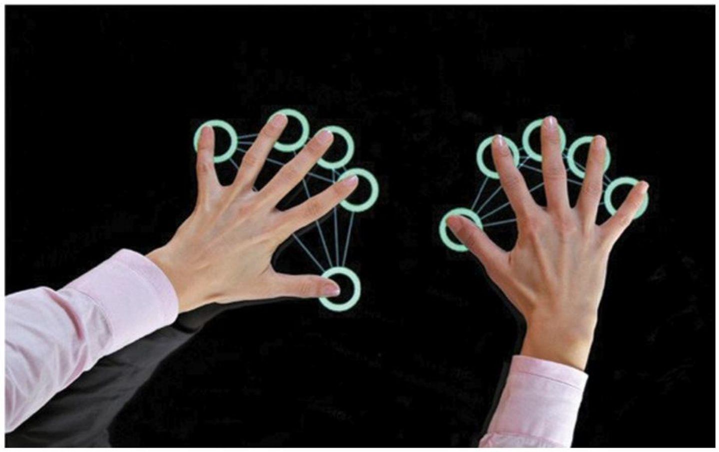


Figure 5.7 A multitouch interface
Source: www.sky-technology.eu/en/blog/article/item/multi-touch-technology-how-it-works.html

- **Gesture-Based Systems**

Gestures involve moving arms and hands to communicate (for instance, waving to say good- bye or raising an arm to speak in class) or to provide information to someone (for example, holding two hands apart to show the size of something). There has been much interest in how technology can be used to capture and recognize a user's gestures for input by tracking them using cameras and then analyzing them using machine learning algorithms.

David Rose (2018) created a video that depicts many sources of inspiration for where gesture is used in a variety of contexts, including those made by cricket umpires, live concert signers for the deaf, rappers, Charlie Chaplin, mime artists, and Italians. His team at IDEO developed a gesture system to recognize a small set of gestures and used these to control a Philips HUE light set and a Spotify station. They found that gestures need to be sequential to be understood in the way a sentence is composed of a noun, then verb, and object plus operation. For example, for "speaker, on," they used a gesture on one hand to designate the noun, and another on the other hand to designate the verb. So, to change the volume, the user needs to point to a speaker with their left hand while raising their right hand to signal turn the volume up.

- **Haptic Interfaces**

Haptic interfaces provide tactile feedback, by applying vibration and forces to the person, using actuators that are embedded in their clothing or a device that they are carrying, such as a smartphone or smartwatch. Gaming consoles have also employed vibration to enrich the experience. For example, car steering wheels that are used with driving simulators can vibrate in various ways to provide the feel of the road. As the driver makes a turn, the steering wheel can be programmed to feel like it is resisting—in the way that a real steering wheel does.

Vibrotactile feedback can also be used to simulate the sense of touch between remote people who want to communicate. Actuators embedded in clothing can be designed to re- create the sensation of a hug or a squeeze by being buzzed on various parts of the body.

Another form of feedback is called *ultrahaptics*, which creates the illusion of touch in midair. It does this by using ultrasound to make three-dimensional shapes and textures that can be felt but not seen by the user (www.ultrahaptics.com). This technique can be used to create the illusion of having buttons and sliders that appear in midair. One potential use is in the automotive industry to replace existing physical buttons and knobs or touch- screens. The ultra-haptic buttons and knobs can be designed to appear next to the driver when needed, for example, when detecting the driver wants to turn down the volume or change the radio station.

- **Multimodal Interfaces**

Multimodal interfaces are intended to enrich user experiences by multiplying the way information is experienced and controlled at the interface through using different modalities, such as touch, sight, sound, and speech (Bouchet and Nigay, 2004). Interface techniques that have been combined for this purpose include speech and gesture, eye-gaze and gesture, haptic and audio output, and pen input and speech (Dumas et al., 2009). The assumption is that multimodal interfaces can support more flexible, efficient, and expressive means of human– computer interaction that are more akin to the multimodal experiences that humans encounter in the physical world. Different input/outputs may be used at the same time, for example, using voice commands and gestures simultaneously to move through a virtual environment, or alternately using speech commands followed by gesturing. The most common combination of technologies used for multimodal interfaces is speech and vision processing (Deng and Huang, 2004). Multimodal interfaces can also be combined with multi-sensor input to enable other aspects of the human body to be tracked. For example, eye gaze, facial expressions, and lip movements can also be tracked to provide data about a user's someone's body. On finding it, it locked onto it and measured the three-dimensional positioning of the key joints in their body. This information was converted into a graphical avatar of the user that could be programmed to move just like them. Many people readily saw them- selves as the avatar and learnt how to play games in this manner.

- **Shareable Interfaces**

Shareable interfaces are designed for more than one person to use. Unlike PCs, laptops, and mobile devices, which are aimed at single users, shareable interfaces typically provide multiple inputs and sometimes allow simultaneous input by collocated groups. These include large wall displays, for example SmartBoards (see Figure 4.8a), where people use their own pens or gestures, and interactive tabletops, where small groups can interact with information being displayed on the surface using their fingertips. Examples of interactive tabletops include Smart's SmartTable and Circle Twelve's DiamondTouch (Dietz and Leigh, 2001; see Figure 4.8b). The DiamondTouch tabletop is unique in that it can distinguish between different users touching the surface concurrently. An array of antennae is embedded in the touch surface and each one transmits a unique signal. Each user has their own receiver embedded in a mat on which they're standing or a chair in which they're sitting. When a user touches the tabletop, very small signals are sent through the user's body to their receiver that identifies which antenna has been touched and sends this to the computer. Multiple users can interact simultaneously with digital content using their fingertips.

(a) (b)

Figure 5.8(a) A SmartBoard in use during a meeting and (b) Mitsubishi's interactive tabletop interface
Source: (a) Used courtesy of SMART Technologies Inc. (b) Mitsubishi Electric Research Labs

An advantage of shareable interfaces is that they provide a large interactional space that can support flexible group working, enabling groups to create content together at the same time. Compared with a co-located group trying to work around a single-user PC or laptop, where typically one person takes control, making it more difficult for others to take part, multiple users can interact with large display. Users can point to and touch the information being displayed, while simultaneously viewing the interactions and having the same shared point of reference (Rogers et al., 2009). There are now a number of tabletop apps that have been developed for museums and galleries which enable visitors to learn about various aspects of the environment.

Another type of shareable interface is software platforms that enable groups of people to work together simultaneously even when geographically apart. Various commercial products now exist that enable multiple remote people to work on the same document at the same time (such as Google Docs and Microsoft Excel). Some enable up to 50 people to edit the same document at the same time with more watching on. These software programs provide various functions, such as synchronous editing, tracking changes, annotating, and commenting.

- **Augmented Reality**

Augmented reality (AR) became an overnight success with the arrival of Pokémon Go in 2016. The smartphone app became an instant hit worldwide. Using a player's smartphone camera and GPS signal, the AR game makes it seem as if virtual Pokémon characters are appearing in the real world—popping up all over the place, such as on buildings, on streets, and in parks. As players walk around a given place, they may be greeted with rustling bits of grass that signal a Pokémon nearby. If they walk closer, a Pokémon may pop up on their smartphone screen, as if by magic, and look as if they are actually in front of them. For example, one might be spotted sitting on a branch of a tree or a garden fence.

AR works by superimposing digital elements, like Pokémons, onto physical devices and objects. Closely related to AR is the concept of *mixed reality*, where views of the real world are combined with views of a virtual environment (Drascic and Milgram, 1996). To begin, augmented reality was mostly a subject of experimentation within medicine, where virtual objects, for example X-rays and scans, were overlaid on part of a patient's body to aid the physician's understanding of what was being examined or operated on.

AR was then used to aid controllers and operators in rapid decision-making. One example is air traffic control, where controllers are provided with dynamic information about the aircraft in their section that is overlaid on a video screen showing real planes landing, taking off, and taxiing. The additional information enables the controllers to identify planes easily, which were difficult to make out—something especially useful in poor weather conditions. Similarly, *head-up displays* (HUDs) are used in military and civil planes to aid pilots when landing during poor weather conditions. A HUD provides electronic directional markers on a fold-down display that appears directly in the field of view of the pilot. A number of high- end cars now provide AR windshield technology, where navigation directions can literally look like they are painted on the road ahead of the driver.

Instructions for building or repairing complex equipment, such as photocopiers and car engines, have also been designed to replace paper-based manuals, where drawings are super- imposed upon the machinery itself, telling the mechanic what to do and where to do it. There are also many AR apps available now for a range of contexts, from education to car navigation, where digital content is overlaid on geographic locations and objects. To reveal the digital information, users open the AR app on a smartphone or tablet and the content appears superimposed on what is viewed through the screen.

- **Wearables**

Wearables are a broad category of devices that are worn on the body. These include smart- watches, fitness trackers, fashion tech, and smart glasses. Since the early experimental days of wearable computing, where Steve Mann (1997) donned head and eye cameras to enable him to record what he saw while also accessing digital information on the move, there have been many innovations and inventions, including Google Glass.

New flexible display technologies, e-textiles, and physical computing (for example, Arduino) provide opportunities to design wearables that people will actually want to wear. Jewelry, caps, glasses, shoes, and jackets have all been the subject of experimentation designed to provide the user with a means of interacting with digital information while on the move in the physical world. Early wearables focused on convenience, enabling people to carry out a task (for example, selecting music) without having to take out and control a handheld device. Examples included a ski jacket with integrated music player controls that enabled the wearer to simply touch a button on their arm with their glove to change a music track. More recent applications have focused on how to combine textiles, electronics, and haptic technologies to promote new forms of communication. For example, CuteCircuit developed the KineticDress, which was embedded with sensors that followed the body of the wearer to capture their movements and interaction with others. These were then displayed through electroluminescent embroidery that covered the external skirt section of the dress. Depending on the amount and speed of the wearer's movement, it changed patterns, displaying the wearer's mood to the audience and creating a magic halo around her.

- <u>**Robots and Drones**</u>

Robots have been around for some time, most notably as characters in science-fiction movies, but they also play an important role as part of manufacturing assembly lines, as remote investigators of hazardous locations (for example, nuclear power stations and bomb disposal), and as search and rescue helpers in disasters (for instance in forest fires) or faraway places (like Mars). *Console interfaces* have been developed to enable humans to control and navigate robots in remote terrains, using a combination of joysticks and keyboard controls together with cameras and sensor-based interactions (Baker et al., 2004). The focus has been on designing interfaces that enable users to steer and move a remote robot effectively with the aid of live video and dynamic maps.

Domestic robots that help with the cleaning and gardening have become popular. Robots are also being developed to help the elderly and disabled with certain activities, such as picking up objects and cooking meals. Pet robots, in the guise of human companions, have been commercialized. Several research teams have taken the "cute and cuddly" approach to designing robots, signaling to humans that the robots are more pet-like than human-like.

Drones are a form of unmanned aircraft that are controlled remotely. They were first used by hobbyists and then by the military. Since then, they have become more affordable, accessible, and easier to fly. As a result, they have begun to be used in a wider range of contexts. These include entertainment, such as carrying drinks and food to people at festivals and par- ties; agricultural applications, such as flying them over vineyards and fields to collect data that is useful to farmers; and helping to track poachers in wildlife parks in Africa. Compared with other forms of data collection, they can fly low and stream photos to a ground station where the images can be stitched together into maps and then used to determine the health of a crop or when is the best time to harvest the crop.

- **Brain–Computer Interfaces**

Brain–computer interfaces (BCI) provide a communication pathway between a person's brain waves and an external device, such as a cursor on a screen or a tangible puck that moves via airflow. The person is trained to concentrate on the task (for example, moving the cursor or the puck). Several research projects have investigated how this technique can be used to assist and augment human cognitive or sensory-motor functions. The way BCIs work is by detecting changes in the neural functioning of the brain. Our brains are filled with neurons that comprise individual nerve cells connected to one another by dendrites and axons. Every time we think, move, feel, or remember something, these neurons become active. Small electric signals rapidly move from neuron to neuron, which to a certain extent can be detected by electrodes that are placed on a person's scalp. The electrodes are embedded in specialized headsets, hairnets, or caps.

Brain–computer interfaces have also been developed to control various games. For example, Brainball was developed as a game to be controlled by players' brain waves in which they compete to control a ball's movement across a table by becoming more relaxed and focused. Other possibilities include controlling a robot and being able to fly a virtual plane. Pioneering medical research, conducted by the BrainGate research group at Brown University, has started using brain-computer interfaces to enable people who are paralyzed to control robots.

- **Smart Interfaces**

The motivation for many new technologies is to make them smart, whether it is a smartphone, smartwatch, smart building, smart home, or smart appliance (for example smart lighting, smart speakers, or virtual assistants). The adjective is often used to suggest that the device has some intelligence and it is connected to the Internet. More generally, smart devices are designed to interact with users and other devices connected to a network, many of which are auto- mated, not requiring users to interact with them directly (Silverio-Fernández et al., 2018). The goal is to make them context-aware, that is, to understand what is happening around them and execute appropriate actions. To achieve this, some have been programmed with AI so that they can learn the context and a user's behavior. Using this intelligence, they then change settings or switch things on according to the user's assumed preferences. An example is the smart Nest thermostat that is designed to learn from a householder's behavior. Rather than make the interface invisible, the designers chose to turn it into an aesthetically pleasing one that could be easily viewed.

Smart buildings have been designed to be more energy efficient, efficient, and cost effective. Architects are motivated to use state-of-the-art sensor technology to control building systems, such as ventilation, lighting, security, and heating. Often, the inhabitants of such buildings are considered to be the ones at fault for wasting energy, as they may leave the lights and heating on overnight when not needed, or they forget to lock a door or window. One benefit of having automated systems take control of building services is to reduce these kinds of human errors—a phrase often used by engineers is to take the human "out of the loop." While some smart buildings and homes have improved how they are managed and cut costs, they can also be frustrating to the user, who sometimes would like to be able to open a window to let fresh air in or raise a blind to let in natural lighting. Taking the human out of the loop means that these operations are no longer available. Windows are locked or sealed, and heating is controlled centrally.

Instead of simply introducing ever more automation that takes the human out of the loop further, another approach is to consider the needs of the inhabitants in conjunction with introducing smart technology.

Lesson 2. Natural User Interfaces and Beyond (Sharp, 2019).

As we have seen, there are many kinds of interfaces that can be used to design for user experiences. The staple for many years was the GUI, then the mobile device interface, followed by touch, and now wearables and smart interfaces. Without question, they have been able to support all manner of user activities. What comes next? Will other kinds of interfaces that are projected to be more natural become more mainstream?

A *natural user interface* (NUI) is designed to allow people to interact with a computer in the same way that they interact with the physical world—using their voice, hands, and bodies. Instead of using a keyboard, mouse, or touchpad (as is the case with GUIs), NUIs enable users to speak to machines, stroke their surfaces, gesture at them in the air, dance on mats that detect feet movements, smile at them to get a reaction, and so on. The naturalness refers to the use of everyday skills humans have developed and learned, such as talking, writ- ing, gesturing, walking, and picking up objects. In theory, they should be easier to learn and map more readily onto how people interact with the world than compared with learning to use a GUI.

Instead of having to remember which function keys to press to open a file, a NUI means a person only has to raise their arm or say “open.” But how natural are NUIs? Is it more natural to say “open” than to flick a switch when you want to open a door? And is it more natural to raise both arms to change a channel on the TV than to press a button on a remote device or tell it what to do by speaking to it? Whether a NUI is natural depends on a number of factors, including how much learning is required, the complexity of the app or device’s interface, and whether accuracy and speed are needed (Norman, 2010). Sometimes a gesture is worth a thousand words. Other times, a word is worth a thousand gestures. It depends on how many functions the system supports.

Consider the sensor-based faucets. The gesture-based interface works mostly (with the exception of people wearing black clothing that cannot be detected) because there are only two functions: (1) turning on the water by waving one’s hands under the tap, and (2) turning off the water by removing them from the sink. Now think about other functions that faucets usually provide, such as controlling water temperature and flow. What kind of a gesture would be most appropriate for changing the temperature and then the flow? Would one decide on the temperature first by raising one’s left arm and the flow by raising one’s right arm? How would someone know when to stop raising their arm to get the right temperature? Would they need to put a hand under the tap to check? But if they put their right hand under the tap, that might that have the effect of decreasing the flow? And when does the system know that the desired temperature and flow has been reached? Would it require having both arms suspended in midair for a few seconds to register that was the desired state? It is a difficult problem on how to provide these choices, and it is probably why sensor-based faucets in public bathrooms all have their temperature and flow set to a default.

Our overview of different interface types in this chapter has highlighted how gestural, voice, and other kinds of NUIs have made controlling input and interacting with digital content easier and more enjoyable, even though sometimes they can be less than perfect. For example, using gestures and whole-body movements have proven to be highly enjoyable as a form of input for computer games and physical exercises. Furthermore, new kinds of gesture, voice, and touch interfaces have made the web and online tools more accessible to those who are visually impaired. For example, the iPhone’s VoiceOver control features have empowered visually impaired individuals to be able to easily send email, use the web, play music, and so on, without having to buy an expensive customized phone or screen reader. Moreover, being able to purchase a regular phone means not being singled out for special treatment. And while some gestures may feel cumbersome for sighted people to learn and use, they may not be so for blind or visually impaired people. The iPhone VoiceOver press and guess feature that reads out what you tap on the screen (for example, "messages," "calendar," "mail: 5 new items") can open up new ways of exploring an application while a three-finger tap can become a natural way to turn the screen off.

An emerging class of human–computer interfaces are those that rely largely on subtle, gradual, and continuous changes triggered by information obtained implicitly from the user together with the use of AI algorithms that are coded to learn about the user's behavior and preferences. These are connected with lightweight, ambient, context-aware, affective, and augmented cognition interfaces (Solovey et al., 2014). Using brain, body, behavioral, and environmental sensors, it is now possible to capture subtle changes in people's cognitive and emotional states in real time. This opens up new doors in human–computer interaction. In particular, it allows for information to be used as both continuous and discrete input, potentially enabling new outputs to match and be updated with what people might want and need at any given time. Adding AI to the mix will also enable a new type of interface to emerge that goes beyond simply being natural and smart—one that allows people to develop new superpowers that will enable them to work synergistically with technology to solve ever-more complex problems and undertake unimaginable feats.

Which Interface?

This module presented an overview of the diversity of interfaces that is now available or currently being researched. There are many opportunities to design for user experiences that are a far cry from those originally developed using the command-based interfaces of the 1980s. An obvious question this raises is which one and how do you design it? In many contexts, the requirements for the user experience that have been identified will determine what kind of interface might be appropriate and what features to include. For example, if a healthcare app is being developed to enable patients to monitor their dietary intake, then a mobile device that has the ability to scan barcodes and/or take pictures of food items that can be compared with a database would be a good interface to use, enabling mobility, effective object recognition, and ease of use. If the goal is to design a work environment to support collocated group decision-making activities, then combining shareable technologies and personal devices that enable people to move fluidly among them would be good to consider using.

But how to decide which interface is preferable for a given task or activity? For example, is multimedia better than tangible interfaces for learning? Is voice effective as a command- based interface? Is a multimodal interface more effective than a single media interface? Are wearable interfaces better than mobile interfaces for helping people find information in foreign cities? How does VR differ from AR, and which is the ultimate interface for playing games? In what way are tangible environments more challenging and captivating than virtual worlds? Will shareable interfaces, such as interactive furniture, be better at supporting communication and collaboration compared with using networked desktop technologies? And so forth. These questions are currently being researched. In practice, which interface is most appropriate, most useful, most efficient, most engaging, most supportive, and so on will depend on the interplay of a number of factors, including reliability, social acceptability, privacy, ethical, and location concerns.

Assessment Task

Answer the activity tasks according to the requirement/s needed.

Activity No..1

Choose any game that you or someone you know to play a lot on a smartphone (for example, Candy Crush Saga, Fortnite, or Minecraft). Consider how the game could be played using different interfaces other than the smartphone's. Select three different interfaces (for instance, tangible, wearable, and shareable) and describe in writing how the game could be redesigned for each of these, taking into account the user group being targeted. For example, the tangible game could be designed for children, the wearable interface for young adults, and the shareable interface for older people.

1. Go through the research and design considerations for each interface and consider whether they are relevant for the game setting and what considerations they raise.
2. Describe a hypothetical scenario of how the game would be played for each of the three interfaces.
3. Consider specific design issues that will need to be addressed. For example, for the share- able surface would it be best to have a tabletop or a wall-based surface? How will the users interact with the game elements for each of the different interfaces—by using a pen, finger- tips, voice, or other input device? How do you turn a single-player game into a multiple player one? What rules would you need to add?

4. Compare the pros and cons of designing the game using the three different interfaces with respect to how it is played on the smartphone.

Summary

This Module provided an overview of the diversity of interfaces that can be designed for user experiences, identifying key design issues and research questions that need to be addressed. It has highlighted the opportunities and challenges that lie ahead for designers and researchers who are experimenting with and developing innovative interfaces. It also explained some of the assumptions behind the benefits of different interfaces—some that are currently supported and others that are still unsubstantiated. The chapter presented a number of interaction techniques that are particularly suited (or not) for a given interface type. It also discussed the dilemmas facing designers when using a particular kind of interface, for example, abstract versus realism, menu selection versus free-form text input, and human-like versus non-human-like. Finally, it presented pointers to specific design guidelines and exemplary systems that have been designed using a given interface.

Points to Remember:

- Many interfaces have emerged post the WIMP/GUI era, including voice, wearable, mobile, tangible, brain-computer, smart, robots, and drones.
- A range of design and research questions need to be considered when deciding which interface to use and what features to include.
    - Natural user interfaces may not be as natural as graphical user interfaces—it depends on the task, user, and context.
    - An important concern that underlies the design of any kind of interface is how information is represented to the user (be it speech, multimedia, virtual reality, augmented reality), so that they can make sense of it with respect to their ongoing activity, for example, playing a game, shopping online, or interacting with a pet robot.
    - Increasingly, new interfaces that are context-aware or monitor people raise ethical issues concerned with what data is being collected and for what is it being used.

References

a. Biele, Cezary, "*Human movements in human-computer interaction (HCI*", Springer, 2022

b. Li, Yang and Hilliges, Otmar, *ed.by Artificial intelligence for human computer interaction: a modern approach*, Springer, 2021

c. Thakur, Nirmalya, and B.D., Parameshachari, ed.by" *Human-computer interaction and beyond : advances towards smart and interconnected environments (part I)",* 2021

d. Jennifer Preece, Helen Sharp, Yvonne Rogers, "*Interaction Design: Beyond Human-Computer Interaction", 5th Edition*, Wiley 2019

e. HCI (human-computer interaction) from https://www.cs.bham.ac.uk/~rxb/Teaching/HCI%20II/intro.html

f. HCI (human-computer interface) from www.Tutorialspoint.com

g. Vijay Kanade, *What Is HCI (Human-Computer Interaction)? Meaning, Importance, Examples, and Goals*: https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-hci/

h. What is Human Computer Interaction? By Simplilearn Last updated on Nov 7, 2023: https://www.simplilearn.com/what-is-human-computer-interaction-article

MODULE 6
DATA GATHERING, ANALYSIS, INTERPRETATION AND PRESENTATION

Introduction

Data is everywhere. Indeed, it is common to hear people say that we are drowning in data because there is so much of it. So, what is data? Data can be numbers, words, measurements, descriptions, comments, photos, sketches, films, videos, or almost anything that is useful for understanding a particular design, user needs, and user behavior. Data can be quantitative or qualitative. For example, the time it takes a user to find information on a web page and the number of clicks to get to the information are forms of quantitative data. What the user says about the web page is a form of qualitative data. But what does it mean to collect these and other kinds of data? What techniques can be used, and how useful and reliable is the data that is collected?

This Module presents some techniques for data gathering that are commonly used in interaction design activities. In particular, data gathering is a central part of discovering requirements and evaluation. Within the requirements activity, data gathering is conducted to collect sufficient, accurate, and relevant data so that design can proceed. Within evaluation, data gathering captures user reactions and their performance with a system or proto- type. All of the techniques that we will discuss can be done with little to no programming or technical skills. Recently, techniques for scraping large volumes of data from online activi- ties, such as Twitter posts, have become available. These and other techniques for managing huge amounts of data, and the implications of their use, are discussed in Chapter 10, “Data at Scale.”

Learning Outcomes

At the end of this module, students should be able to:

1. Discuss how to plan and run a successful data gathering program.
2. Enable you to plan and run an interview.

3. Empower you to design a simple questionnaire.
4. Enable you to plan and carry out an observation. Discuss the goal of designing rules for interactive systems.
5. Lists and explain the abstract principles in a general sense of usability.
6. Explain and apply the design rules in establishing standards and following guidelines.
7. Identify, lists and apply the essential characteristics of good design through the 'golden rules or heuristics.

Lesson 1. Data Gathering

Three main techniques for gathering data are introduced in this chapter: interviews, questionnaires, and observation. The next chapter discusses how to analyze and interpret the data collected. Interviews involve an interviewer asking one or more interviewees a set of questions, which may be highly structured or unstructured; interviews are usually synchronous and are often face-to-face, but they don't have to be. Increasingly, interviews are conducted remotely using one of the many teleconferencing systems, such as Skype or Zoom, or on the phone. Questionnaires are a series of questions designed to be answered asynchronously, that is, without the presence of the investigator. These questionnaires may be paper-based or available online. Observation may be direct or indirect. Direct observation involves spending time with individuals observing their activities as they happen. Indirect observation involves making a record of the user's activity as it happens, to be studied at a later date. All three techniques may be used to collect qualitative or quantitative data.

Although this is a small set of basic techniques, they are flexible and can be combined and extended in many ways. Indeed, it is important not to focus on just one data gathering technique, if possible, but to use them in combination so as to avoid biases that are inherent in any one approach.

Lesson 1.1 Five Key Issues (Sharp, 2019).

Five key issues require attention for any data gathering session to be successful: goal setting, identifying participants, the relationship between the data collector and the data provider, triangulation, and pilot studies.

- **Setting Goals**

The main reason for gathering data is to glean information about users, their behavior, or their reaction to technology. Examples include understanding how technology fits into family life, identifying which of two icons representing "send message" is easier to use, and finding out whether the planned redesign for a handheld meter reader is headed in the right direction. There are many different reasons for gathering data, and before beginning, it is important to set specific goals for the study. These goals will influence the nature of data gathering sessions, the data gathering techniques to be used, and the analysis to be performed (Robson and McCartan, 2016).

The goals may be expressed more or less formally, for instance, using some structured or even mathematical format or using a simple description such as the ones in the previous paragraph. Whatever the format, however, they should be clear and concise. In interaction design, it is more common to express goals for data gathering informally.

- **Identifying Participants**

The goals developed for the data gathering session will indicate the types of people from whom data is to be gathered. Those people who fit this profile are called the *population* or *study population*. In some cases, the people from whom to gather data may be clearly identifiable—maybe because there is a small group of users and access to each one is easy. However, it is more likely that the participants to be included in data gathering need to be chosen, and this is called *sampling*. The situation where all members of the target population are accessible is called *saturation sampling*, but this is quite rare. Assuming that only a portion of the population will be involved in data gathering, then there are two options: prob- ability sampling or nonprobability sampling. In the former case, the most commonly used approaches are simple random sampling or stratified sampling; in the latter case, the most common approaches are convenience sampling or volunteer panels.

Random sampling can be achieved by using a random number generator or by choosing every *n*th person in a list. Stratified sampling relies on being able to divide the population into groups (for example, classes in a secondary school) and then applying random sampling. Both convenience sampling and volunteer panels rely less on choosing the participants and more on the participants being prepared to take part. The term *convenience sampling* is used to describe a situation where the sample includes those who were available rather than those specifically selected. Another form of convenience sampling is *snowball sampling*, in which a current participant finds another participant and that participant finds another, and so on. Much like a snowball adds more snow as it gets bigger, the population is gathered up as the study progresses. The crucial difference between probability and nonprobability methods is that in the former you can apply statistical tests and generalize to the whole population, while in the latter such generalizations are not robust. Using statistics also requires a sufficient number of participants.

- **Relationship with Participants**

One significant aspect of any data gathering is the relationship between the person (people) doing the gathering and the person (people) providing the data. Making sure that this relationship is clear and professional will help to clarify the nature of the study. How this is achieved varies in different countries and different settings. In the United States and United Kingdom, for example, it is achieved by asking participants to sign an informed consent form, while in Scandinavia such a form is not required. The details of this form will vary, but it usually asks the participants to confirm that the purpose of the data gathering and how the data will be used has been explained to them and that they are willing to continue. It usually explains that their data will be private and kept in a secure place. It also often includes a statement that participants may withdraw at any time and that in this case none of their data will be used in the study.

The informed consent form is intended to protect the interests of both the data gatherer and the data provider. The gatherer wants to know that the data they collect can be used in their analysis, presented to interested parties, and published in reports. The data provider wants reassurance that the information they give will not be used for other purposes or in any context that would be detrimental to them. For example, they want to be sure that personal contact information and other personal details are not made public. This is especially true when people with disabilities or children are being interviewed. In the case of children, using an informed consent form reassures parents that their children will not be asked threatening, inappropriate, or embarrassing questions, or be asked to look at disturbing or violent images. In these cases, parents are asked to sign the form. Figure 6.1 shows an example of a typical informed consent form.

This kind of consent is also not generally required when gathering requirements data for a commercial company where a contract usually exists between the data collector and the data provider. An example is where a consultant is hired to gather data from company staff during the course of discovering requirements for a new interactive system to support timesheet entry. The employees of this company would be the users of the system, and the consultant would therefore expect to have access to the employees to gather data about the timesheet activity. In addition, the company would expect its employees to cooperate in this exercise. In this case, there is already a contract in place that covers the data gathering activity, and therefore an informed consent form is less likely to be required. As with most ethical issues, the important thing is to consider the situation and make a judgment based on the specific circumstances. Increasingly, projects and organizations that collect personal data from

Crowdsourcing Design for Citizen Science Organizations

SHORT VERSION OF CONSENT FORM for participants at the University of Maryland – 18 YEARS AND OLDER

You are invited to participate in a research project being conducted by the researchers listed on the bottom of the page. In order for us to be allowed to use any data you wish to provide, we must have your consent.
In the simplest terms, we hope you will use the mobile phone, tabletop, and project website at the University of Maryland to

- Take pictures
- Share observations about the sights you see on campus
- Share ideas that you have to improve the design of the phone or tabletop application or website
- Comment on pictures, observations, and design ideas of others

The researchers and others using CampusNet will be able to look at your comments and pictures on the tabletop and/or website, and we may ask if you are willing to answer a few more questions (either on paper, by phone, or face-to-face) about your whole experience. You may stop participat- ing at any time.

A long version of this consent form is available for your review and signature, or you may opt to sign this shorter one by *checking off all the boxes that reflect your wishes and signing and dating the form below.*

____I agree that any photos I take using the CampusNet application may be uploaded to the tabletop at the University of Maryland and/or a website now under development.
____I agree to allow any comments, observations, and profile information that I choose to share with others via the online application to be visible to others who use the application at the same time or after me.
____I agree to be videotaped/audiotaped during my participation in this study.
____I agree to complete a short questionnaire during or after my participation in this study.

NAME [Please print]	
SIGNATURE	
DATE	

[Contact information of Senior Researcher responsible for the project] people need to demonstrate that it is protected from unauthorized access.

Incentives to take part in data gathering sessions may also be needed. For example, if there is no clear advantage to the respondents, incentives may persuade them to take part; in other circumstances, respondents may see it as part of their job or as a course requirement to take part. In this case, the motivation for providing the required information is clear. However, when collecting data to understand how appealing a new interactive app is for school children, different incentives would be appropriate. Here, the advantage for individuals to take part is not so obvious.

➢ Triangulation

Triangulation is a term used to refer to the investigation of a phenomenon from (at least) two different perspectives (Denzin, 2006). Four types of triangulation have been defined (Jupp, 2006).

- Triangulation of data means that data is drawn from different sources at different times, in different places, or from different people (possibly by using a different sampling technique).
- Investigator triangulation means that different researchers (observers, interviewers, and so on) have been involved in collecting and interpreting the data.
- Triangulation of theories means the use of different theoretical frameworks through which to view the data or findings.
- Methodological triangulation means to employ different data gathering techniques.

The last of these is the most common form of triangulation—to validate the results of some inquiry by pointing to similar results yielded through different perspectives. However, validation through true triangulation is difficult to achieve. Different data gathering methods result in different kinds of data, which may or may not be compatible. Using different theoretical frameworks may or may not result in complementary findings, but to achieve theoretical triangulation would require the theories to have similar philosophical underpinnings. Using more than one data gathering technique, and more than one data analysis approach, is good practice because it leads to insights from the different approaches even though it may not be achieving true triangulation.

➢ Pilot Studies

A pilot study is a small trial run of the main study. The aim is to make sure that the proposed method is viable before embarking on the real study. For example, the equipment and instructions can be checked, the questions for an interview or in a questionnaire can be tested for clarity, and an experimental procedure can be confirmed as viable. This can identify potential problems in advance so that they can be corrected. Distributing 500 questionnaires and then being told that two of the questions were very confusing wastes time, annoys participants, and is an expensive error that could be avoided by doing a pilot study.

If it is difficult to find participants or access to them is limited, asking colleagues or peers to participate can work as an alternative for a pilot study. Note that anyone involved in a pilot study cannot be involved in the main study itself. Why? Because they will know more about the study and this can distort the results.

Lesson 1.2 Data Recording (Sharp, 2019).

Capturing data is necessary so that the results of a data gathering session can be analyzed and shared. Some forms of data gathering, such as questionnaires, diaries, interaction logging, scraping, and collecting work artifacts, are self-documenting and no further recording is necessary. For other techniques, however, there is a choice in recording approaches. The most common of these are taking notes, photographs, or recording audio or video. Often, several data recording approaches are used together. For example, an interview may be voice recorded, and then to help the interviewer in later analysis, a photograph of the interviewee may be taken to remind the interviewer about the context of the discussion.

Which data recording approaches are used will depend on the goal of the study and how the data will be used, the context, the time and resources available, and the sensitivity of the situation; the choice of data recording approach will affect the level of detail collected and how intrusive the data gathering will be. In most settings, audio recording, photographs, and notes will be sufficient. In others, it is essential to collect video data so as to record in detail the intricacies of the activity and its context. Three common data recording approaches are discussed next.

- **Notes Plus Photographs**

Taking notes (by hand or by typing) is the least technical and most flexible way of recording data, even if it seems old-fashioned. Handwritten notes may be transcribed in whole or in part, and while this may seem tedious, it is usually the first step in analysis, and it gives the analyst a good overview of the quality and contents of the data collected. Tools exist for supporting data collection and analysis, but the advantages of handwritten notes include that using pen and paper can be less intrusive than typing and is more flexible, for example, for drawing diagrams of work layouts. Furthermore, researchers often comment that writ- ing notes helps them to focus on what is important and starts them thinking about what the data is telling them. The disadvantages of notes include that it can be difficult to capture the right highlights, and it can be tiring to write and listen or observe at the same time. It is easy to lose concentration, biases creep in, handwriting can be difficult to decipher, and the speed of writing is limited. Working with a colleague can reduce some of these problems while also providing another perspective.

If appropriate, photograph(s) and short videos (captured via smartphones or other hand-held devices) of artifacts, events, and the environment can supplement notes and hand-drawn sketches, providing that permission has been given to collect data using these approaches.

- **Audio Plus Photographs**

Audio recording is a useful alternative to note-taking and is less intrusive than video. During observation, it allows observers to focus on the activity rather than on trying to capture every spoken word. In an interview, it allows the interviewer to pay more attention to the interviewee rather than trying to take notes as well as listening. It isn’t always necessary to transcribe all of the data collected—often only sections are needed, depending on the goals of the study. Many studies do not need a great level of detail, and instead recordings are used as a reminder and as a source of anecdotes for reports. It is surprising how evocative audio recordings of people or places from the data session can be, and those memories provide added context to the analysis. If audio recording is the main or only data collection technique, then the quality needs to be good; performing interviews remotely, for example using Skype, can be compromised because of poor connections and acoustics. Audio recordings are often supplemented with photographs.

- **Video**

Smartphones can be used to collect short video clips of activity. They are easy to use and less obtrusive than setting up sophisticated cameras. But there are occasions when a video is needed for long periods of time or when holding a phone is unreliable, for example, recording how designers collaborate together in a workshop or how teens interact in a “makerspace,” in which people can work on projects while sharing ideas, equipment, and knowledge. For these kinds of sessions, more professional video equipment that clearly captures both visual and audio data is more appropriate. Other ways of recording facial expressions together with verbal comments are also being used, which can be operated both in-person and remotely. When considering whether to use a camera, Heath et al. (2010) suggest the following issues to consider:

- *Deciding whether to fix the camera’s position or use a roving recorder*. This decision depends on the activity being recorded and the purpose to which the video data will be put, for example, for illustrative purposes only or for detailed data analysis. In some cases, such as pervasive games, a roving camera is the only way to capture the required action. For some studies, the video on a smartphone may be adequate and require less effort to set up.
- *Deciding where to point the camera in order to capture what is required*. Heath and his colleagues suggest carrying out fieldwork for a short time before starting to video record in order to become familiar with the environment and be able to identify suitable recording locations. Involving the participants themselves in deciding what and where to record also helps to capture relevant action.
- *Understanding the impact of the recording on participants*. It is often assumed that video recording will have an impact on participants and their behavior. However, it is worth taking an empirical approach to this issue and examining the data itself to see whether there is any evidence of people changing their behavior such as orienting themselves toward the camera.

Lesson 1.3 Interviews (Sharp, 2019).

Interviews can be thought of as a “conversation with a purpose” (Kahn and Cannell, 1957). How much like an ordinary conversation the interview will be depends on the type of interview. There are four main **types of interviews**: open-ended or unstructured, structured, semi-structured, and group interviews (Fontana and Frey, 2005). The first three types are named according to how much control the interviewer imposes on the conversation by following a predetermined set of questions. The fourth type, which is often called a *focus group*, involves a small group guided by a facilitator. The facilitation may be quite informal or follow a structured format.

The most appropriate approach to interviewing depends on the purpose of the interview, the questions to be addressed, and the interaction design activity. For example, if the goal is first to gain impressions about users' reactions to a new design concept, then an informal, open- ended interview is often the best approach. But if the goal is to get feedback about a particular design feature, such as the layout of a new web browser, then a structured interview or questionnaire is often better. This is because the goals and questions are more specific in the latter case.

- **Unstructured Interviews**

  Open-ended or unstructured interviews are at one end of a spectrum of how much control the interviewer has over the interview process. They are exploratory and are similar to conversations around a particular topic; they often go into considerable depth. Questions posed by the interviewer are open, meaning that there is no particular expectation about the format or content of answers.

- **Structured Interviews**

  In structured interviews, the interviewer asks predetermined questions similar to those in a questionnaire and the same questions are used with each participant so that the study is standardized. The questions need to be short and clearly worded, and they are typically closed questions, which means that they require an answer from a predetermined set of alternatives. (This may include an "other" option, but ideally this would not be chosen often.) Closed questions work well if the range of possible answers is known or if participants don't have much time. Structured interviews are useful only when the goals are clearly understood and specific questions can be identified. Example questions for a structured interview might be the following:

  - "Which of the following websites do you visit most frequently: Amazon.com, Google.com, or msn.com?"
  - "How often do you visit this website: every day, once a week, once a month, less often than once a month?"
  - "Do you ever purchase anything online: Yes/No? If your answer is Yes, how often do you

purchase things online: every day, once a week, once a month, less frequently than once a month?"

Questions in a structured interview are worded the same for each participant and are asked in the same order.

- **Semi-structured Interviews**

Semi-structured interviews combine features of structured and unstructured interviews and use both closed and open questions. The interviewer has a basic script for guidance so that the same topics are covered with each interviewee. The interviewer starts with preplanned questions and then probes the interviewee to say more until no new relevant information is forthcoming. Here's an example:

> ***Which music websites do you visit most frequently?***
> ***Answer: Mentions several but stresses that they prefer hottestmusic.com Why?***
> ***Answer: Says that they like the site layout Tell me more about the site layout.***
> ***Answer: Silence, followed by an answer describing the site's layout Anything else that you like about the site?***
> ***Answer: Describes the animations***
> ***Thanks. Are there any other reasons for visiting this site so often that you haven't mentioned?***

It is important not to pre-empt an answer by phrasing a question to suggest that a particular answer is expected. For example, "You seemed to like this use of color . . ." assumes that this is the case and will probably encourage the interviewee to answer that this is true so as not to offend the interviewer. Children are particularly prone to behave in this way. The body language of the interviewer, for example whether they are smiling, scowling, looking disapproving, and so forth, can have a strong influence on whether the interviewee will agree with a question, and the interviewee needs to have time to speak and not be rushed.

Probes are a useful device for getting more information, especially neutral probes such as "Do you want to tell me anything else?" and prompts that remind interviewees if they forget terms or names help to move the interview along. Semi-structured interviews are intended to be broadly replicable, so probing and prompting aim to move the interview along without introducing bias.

- **Focus Groups**

  Interviews are often conducted with one interviewer and one interviewee, but it is also com- mon to interview people in groups. One form of group interview that is sometimes used in interaction design activities is the focus group. Normally, three to ten people are involved, and the discussion is led by a trained facilitator. Participants are selected to provide a representative sample of the target population. For example, in the evaluation of a university website, a group of administrators, faculty, and students may form three separate focus groups because they use the web for different purposes. In requirements activities, a focus group may be held in order to identify conflicts in expectations or terminology from different stakeholders.

  The benefit of a focus group is that it allows diverse or sensitive issues to be raised that might otherwise be missed, for example in the requirements activity to understand multiple points within a collaborative process or to hear different user stories (Unger and Chandler, 2012). The method is more appropriate for investigating shared issues rather than individual experiences. Focus groups enable people to put forward their own perspectives. A preset agenda is developed to guide the discussion, but there is sufficient flexibility for the facilitator to follow unanticipated issues as they are raised. The facilitator guides and prompts discussion, encourages quiet people to participate, and stops verbose ones from dominating the discussion. The discussion is usually recorded for later analysis, and participants may be invited to explain their comments more fully at a later date.

Lesson 1.4 Planning and Conducting an Interview

Planning an interview involves developing the set of questions or topics to be covered, collating any documentation to give to the interviewee (such as consent form or project description), checking that recording equipment works, structuring the interview, and organizing a suitable time and place.

- **Developing Interview Questions**

  Questions may be open-ended (or open) or closed-ended (or closed). Open questions are best suited where the goal of the session is exploratory; closed questions are best suited where the possible answers are known in advance. An unstructured interview will usually consist mainly of open questions, while a structured interview will usually consist of closed questions. A semi-structured interview may use a combination of both types.

The following guidelines help in developing interview questions (Robson and McCartan, 2016):

- Long or compound questions can be difficult to remember or confusing, so split them into two separate questions. For example, instead of “How do you like this smartphone app compared with previous ones that you have used?” say, “How do you like this smartphone app?” “Have you used other smartphone apps?” If so, “How did you like them?” This is easier for the interviewee to respond to and easier for the interviewer to record.
- Interviewees may not understand jargon or complex language and might be too embarrassed to admit it, so explain things to them in straightforward ways.
- Try to keep questions neutral, both when preparing the interview script and in conversation during the interview itself. For example, if you ask “Why do you like this style of interaction?” this question assumes that the person does like it and will discourage some interviewees from stating their real feelings.

- **Running the Interview**

Before starting, make sure that the goals of the interview have been explained to the inter- viewee and that they are willing to proceed. Finding out about the interviewee and their environment before the interview will make it easier to put them at ease, especially if it is an unfamiliar setting.

During the interview, it is better to listen more than to talk, to respond with sympathy but without bias, and to appear to enjoy the interview. The following is a common sequence for an interview (Robson and McCartan, 2016):

1. An introduction in which the interviewer introduces themselves and explains why they are doing the interview, reassures interviewees regarding any ethical issues, and asks if they mind being recorded, if appropriate. This should be exactly the same for each interviewee.

2. A warm-up session where easy, nonthreatening questions come first. These may include questions about demographic information, such as “What area of the country do you live in?”
3. A main session in which the questions are presented in a logical sequence, with the more probing ones at the end. In a semi-structured interview, the order of questions may vary between participants, depending on the course of the conversation, how much probing is done, and what seems more natural.
4. A cooling-off period consisting of a few easy questions (to defuse any tension that may have arisen).
5. A closing session in which the interviewer thanks the interviewee and switches off the recorder or puts their notebook away, signaling that the interview has ended.

➢ Other Forms of Interview

Conducting face-to-face interviews and focus groups can be impractical, but the prevalence of Skype, Cisco WebEx, Zoom, and other digital conferencing systems, email, and phone- based interactions (voice or chat), sometimes with screen-sharing software, make remote interviewing a good alternative. These are carried out in a similar fashion to face-to-face sessions, but poor connections and acoustics can cause different challenges, and participants may be tempted to multitask rather than focus on the session at hand. Advantages of remote focus groups and interviews, especially when done through audio-only channels, include the following:

- The participants are in their own environment and are more relaxed.
- Participants don’t have to travel.
- Participants don’t need to worry about what they wear.
- For interviews involving sensitive issues, interviewees can remain anonymous.

In addition, participants can leave the conversation whenever they want to by just cutting the connection, which adds to their sense of security. From the interviewer’s perspective, a wider set of participants can be reached easily, but a potential disadvantage is that the facilitator does not have a good view of the interviewees’ body language.

For more information and some interesting thoughts on remote usability testing, see ***http://www.uxbooth.com/articles/hidden-benefits-remote-research/***

Retrospective interviews, that is, interviews that reflect on an activity or a data gathering session in the recent past, may be conducted with participants to check that the interviewer has correctly understood what was happening. This is a common practice in observational studies where it is sometimes referred to as *member checking*.

Lesson 1.5 Questionnaires (Sharp, 2019).

Questionnaires are a well-established technique for collecting demographic data and users' opinions. They are similar to interviews in that they can have closed or open-ended questions, but once a questionnaire is produced, it can be distributed to a large number of participants without requiring additional data gathering resources. Thus, more data can be collected than would normally be possible in an interview study. Furthermore, participants who are located in remote locations or those who cannot attend an interview at a particular time can be involved more easily. Often a message is sent electronically to potential participants directing them to an online questionnaire.

Effort and skill are needed to ensure that questions are clearly worded and the data collected can be analyzed efficiently. Well-designed questionnaires are good for getting answers to specific questions from a large group of people. Questionnaires can be used on their own or in conjunction with other methods to clarify or deepen understanding. For example, information obtained through interviews with a small selection of interviewees might be corroborated by sending a questionnaire to a wider group to confirm the conclusions.

Questionnaire questions and structured interview questions are similar, so which technique is used when? Essentially, the difference lies in the motivation of the respondent to answer the questions. If their motivation is high enough to complete a questionnaire without anyone else present, then a questionnaire will be appropriate. On the other hand, if the respondents need some persuasion to answer the questions, a structured interview format would be better. For example, structured interviews are easier and quicker to conduct if people will not stop to complete a questionnaire, such as at a train station or while walking to their next meeting.

It can be harder to develop good questionnaire questions compared with structured interview questions because the interviewer is not available to explain them or to clarify any ambiguities. Because of this, it is important that questions are specific; when possible, ask closed-ended questions and offer a range of answers, including a “no opinion” or “none of these” option. Finally, use negative questions carefully, as they can be confusing and may lead to false information. Some questionnaire designers, however, use a mixture of negative and positive questions deliberately because it helps to check the users’ intentions.

➢ Questionnaire Structure

Many questionnaires start by asking for basic demographic information (gender, age, place of birth) and details of relevant experience (the number of hours a day spent searching on the Internet, the level of expertise within the domain under study, and so on). This background information is useful for putting the questionnaire responses into context. For example, if two responses conflict, these different perspectives may be because of their level of experience—a group of people who are using a social networking site for the first time are likely to express different opinions than another group with five years’ experience of using such sites. However, only contextual information that is relevant to the study goal needs to be collected. For example, it is unlikely that a person’s height will provide relevant context to their responses about Internet use, but it might be relevant for a study concerning wearables.

Specific questions that contribute to the data-gathering goal usually follow these demo- graphic questions. If the questionnaire is long, the questions may be subdivided into related topics to make it easier and more logical to complete.

The following is a checklist of general advice for designing a questionnaire:

- Think about the ordering of questions. The impact of a question can be influenced by question order.
- Consider whether different versions of the questionnaire are needed for different populations.
- Provide clear instructions on how to complete the questionnaire, for example, whether answers can be saved and completed later. Aim for both careful wording and good typography.
- Think about the length of the questionnaire, and avoid questions that don’t address the study goals.
- If the questionnaire has to be long, consider allowing respondents to opt out at

different stages. It is usually better to get answers to some sections than no answers at all because of dropout.

- Think about questionnaire layout and pacing; for instance, strike a balance between using white space, or individual web pages, and the need to keep the questionnaire as compact as possible.

➢ Question and Response Format

Different formats of question and response can be chosen. For example, with a closed-ended question, it may be appropriate to indicate only one response, or it may be appropriate to indicate several. Sometimes, it is better to ask users to locate their answer within a range. Selecting the most appropriate question and response format makes it easier for respondents to answer clearly. Some commonly used formats are described next.

- **Check Boxes and Ranges**

  The range of answers to demographic questions is predictable. Nationality, for example, has a finite number of alternatives, and asking respondents to choose a response from a predefined list makes sense for collecting this information. A similar approach can be adopted if details of age are needed. But since some people do not like to give their exact age, many questionnaires ask respondents to specify their age as a range.

  A frequently asked question about ranges is whether the interval must be equal in all cases. The answer is no—it depends on what you want to know. For example, people who might use a website about life insurance are likely to be employed individuals who are 21–65 years old. The question could, therefore, have just three ranges: under 21, 21–65, and over 65. In contrast, to see how the population's political views vary across generations might require 10-year groups for people over 21, in which case the following ranges would be appropriate: under 21, 21–30, 31–40, and so forth.

- **Rating Scales**

  There are a number of different types of rating scales, each with its own purpose. Two commonly used scales are the ***Likert*** and ***semantic differential*** scales. Their purpose is to elicit a range of responses to a question that can be compared across respondents. They are good for getting people to make judgments, such as how easy, how usable, and the like. Likert scales rely on identifying a set of statements representing a range of possible opinions, while semantic differential scales rely on choosing pairs of words that represent the range of possible opinions. Likert scales are more commonly used because identifying suitable statements that respondents will understand consistently is easier than identifying semantic pairs that respondents interpret as intended.

Likert Scales

Likert scales are used for measuring opinions, attitudes, and beliefs, and consequently they are widely used for evaluating user satisfaction with products. For example, users' opinions about the use of color in a website could be evaluated with a Likert scale using a range of numbers, as in question 1 here, or with words as in question 2:

1) The use of color is excellent (where 1 represents strongly agree and 5 represents strongly disagree):

1	2	3	4	5
☐	☐	☐	☐	☐

2) The use of color is excellent:

Strongly agree	Agree	OK	Disagree	Strongly disagree
☐	☐	☐	☐	☐

In both cases, respondents would be asked to choose the right box, number, or phrase.

Designing a Likert scale involves the following steps:

1. *Gather a pool of short statements about the subject to be investigated*. Examples are “This control panel is clear” and “The procedure for checking credit rating is too complex. ” A brainstorming session with peers is a good way to identify key aspects to be investigated.
2. *Decide on the scale*. There are three main issues to be addressed here: How many

points does the scale need? Should the scale be discrete or continuous? How can the scale be rep- resented? See Box 8.4 What Scales to Use: Three, Five, Seven, or More? for more on this.

3. *Select items for the final questionnaire, and reword as necessary to make them clear.*

In the first example above, the scale is arranged with 1 as the highest choice on the left and 5 as the lowest choice on the right. The logic for this is that first is the best place to be in a race and fifth would be the worst place. While there is no absolute right or wrong way of ordering the numbers other researchers prefer to arrange the scales the other way around with 1 as the lowest on the left and 5 as the highest on the right. They argue that intuitively the highest number suggests the best choice and the lowest number suggests the worst choice. Another reason for going from lowest to highest is that when the results are reported, it is more intuitive for readers to see high numbers representing the best choices. The important thing is to be consistent.

Semantic Differential Scales

Semantic differential scales explore a range of bipolar attitudes about a particular item, each of which is represented as a pair of adjectives. The participant is asked to choose a point between the two extremes to indicate agreement with the poles, as shown in Figure 8.5. The score for the investigation is found by summing the scores for each bipolar pair. Scores are then computed across groups of participants. Notice that in this example the poles are mixed so that good and bad features are distributed on the right and the left. In this example, there are seven positions on the scale.

Attractive	|___|___|___|___|___|___|___|	Ugly
Clear		Confusing
Dull		Colorful
Exciting		Boring
Annoying		Pleasing
Helpful		Unhelpful
Poor		Well designed

Figure 5.1 An example of a semantic differential scale

Administering Questionnaires

Two important issues when using questionnaires are reaching a representative sample of participants and ensuring a reasonable response rate. For large surveys, potential respondents need to be selected using a sampling technique. However, interaction designers commonly use a small number of participants, often fewer than 20 users. Completion rates of 100 percent are often achieved with these small samples, but with larger or more remote populations, ensuring that surveys are returned is a well-known problem. A 40 percent return is generally acceptable for many surveys, but much lower rates are common. Depending on your audience, you might want to consider offering incentives.

While questionnaires are often online, paper questionnaires may be more convenient in some situations, for example, if participants do not have Internet access or if it is expensive to use. Occasionally, short questionnaires are sent within the body of an email, but more often the advantages of the data being compiled automatically and either partly or fully analyzed make online questionnaires attractive. Online questionnaires are interactive and can include check boxes, radio buttons, pull-down and pop-up menus, help screens, graphics, or videos. They can also provide immediate data validation; for example, the entry must be a number between 1 and 20, and automatically skip questions that are irrelevant to some respondents, such as questions aimed only at teenagers. Other advantages of online questionnaires include faster response rates and automatic transfer of responses into a database for analysis

The main problem with online questionnaires is the difficulty of obtaining a random sample of respondents; online questionnaires usually rely on convenience sampling, and hence their results cannot be generalized. In some countries, online questions, often delivered via smartphones, are frequently used in conjunction with television to elicit viewers' opinions of programs and political events.

Deploying an online questionnaire involves the following steps

1. *Plan the survey timeline*. If there is a deadline, work backward from the deadline and plan what needs to be done on a weekly basis.
2. *Design the questionnaire offline*. Using plain text is useful as this can then be copied more easily into the online survey tool.
3. *Program the online survey*. How long this will take depends on the complexity of the design,

for example, how many navigational paths it contains or if it has a lot of interactive features.

4. *Test the survey, both to make sure that it behaves as envisioned and to check the questions themselves*. This includes getting feedback from content experts, survey experts, and potential respondents. This last group forms the basis of a pilot study.
5. *Recruit respondents*. As mentioned earlier, participants may have different reasons for taking part in the survey, but especially when respondents need to be encouraged, make the invitations intriguing, simple, friendly, respectful, trustworthy, motivating, interesting, informative, and short.

There are many online questionnaire templates available that provide a range of options, including different question types (for example open-ended, multiple choice), rating scales (such as Likert, semantic differential), and answer types (for example, radio buttons, check boxes, drop-down menus).

The following activity asks you to make use of one of these templates. Apart from being able to administer an online questionnaire widely, these templates also enable the questionnaire to be segmented. For example, airline satisfaction questionnaires often have different sections for check-in, baggage handling, airport lounge, inflight movies, inflight food service, and so forth. If you didn't use an airport lounge or check your baggage, you can skip those sections. This avoids respondents getting frustrated by having to go through questions that are not relevant to them. It is also a useful technique for long questionnaires, as it ensures that if a respondent opts out for lack of time or gets tired of answering the questions, the data that has been provided already is available to be analyzed.

Lesson 1.6 Choosing and Combining Techniques (Sharp, 2019).

Combining data gathering techniques into a single data gathering program is common practice, for example, when collecting case study data. The benefit of using a combination of methods is to provide multiple perspectives. Choosing which data gathering techniques to use depends on a variety of factors related to the study goals. There is no right technique or combination of techniques, but some will undoubtedly be more appropriate than others. The decision about which to use will need to be made after taking all of the factors into account.

Table 5.1 provides an overview to help choose a set of techniques for a specific project. It lists the kind of information obtained (such as answers to specific questions) and the type of data (for example, mostly qualitative or mostly quantitative). It also includes some advantages and disadvantages for each technique. Note that different modalities can be used for some of these techniques. For example, interviews and focus groups can be conducted face-to-face, by phone, or through teleconferencing, so when considering advantages and disadvantages of the techniques, this should also be taken into account.

In addition, technique choice is influenced by practical issues.

- *The focus of the study*. What kind of data will support the focus and goal of the study? This will be influenced by the interaction design activity and the level of maturity of the design.
- *The participants involved*. Characteristics of the target user group including their location and availability.
- *The nature of the technique*. Does the technique require specialist equipment or training, and do the investigators have the appropriate knowledge and experience?
- *Available resources*. Expertise, tool support, time, and money.

Table 5.1 Overview of data gathering techniques and their use

Technique	Good for	Kind of data	Advantages	Disadvantages
Interviews	Exploring issues	Some quantitative but mostly qualitative	Interviewer can guide interviewee if necessary. Encourages contact between developers and users.	Artificial environment may intimidate interviewee. It also removes them from the environment where work is typically being done.
Focus groups	Collecting multiple viewpoints	Some quantitative but mostly qualitative	Highlights areas of consensus and conflict. Encourages contact between developers and users.	Possibility of dominant characters.
Questionnaires	Answering specific questions	Quantitative and qualitative	Can reach many people with low resource requirements.	The design is key. Response rates may be low. Unless carefully designed, the responses may not provide suitable data.
Direct observation in the field	Understanding context of user activity	Mostly qualitative	Observing gives insights that other techniques don't provide.	Very time-consuming. Huge amounts of data are produced.
Direct observation in a controlled environment	Capturing the detail of what individuals do	Quantitative and qualitative	Can focus on the details of a task without interruption.	Results may have limited use in the normal environment because the conditions were artificial.
Indirect observation	Observing users without disturbing their activity; data captured automatically	Quantitative (logging) and qualitative (diary)	User doesn't get distracted by the data gathering; automatic recording means that it can extend over long periods of time.	A large amount of quantitative data needs tool support to analyze (logging); participants' memories may exaggerate (diary).

Lesson 2. Quantitative and Qualitative (Sharp, 2019).

Quantitative data is in the form of numbers, or data that can easily be translated into numbers. Examples are the number of years' experience the interviewees have, the number of projects a department handles at a time, or the number of minutes it takes to perform a task.

Qualitative data is in the form of words and images, and it includes descriptions, quotes from interviewees, vignettes of activity, and photos. It is possible to express qualitative data in numerical form, but it is not always **meaningful to do so.**

It is sometimes assumed that certain forms of data gathering can only result in quantitative data and that others can only result in qualitative data. However, this is a fallacy. All forms of data gathering discussed in the previous chapter may result in qualitative and quantitative data. For example, on a questionnaire, questions about the participant's age or number of software apps they use in a day will result in quantitative data, while any comments will result in qualitative data. In an observation, quantitative data that may be recorded includes the number of people involved in a project or how many hours someone spends sorting out a problem, while notes about feelings of frustration, or the nature of interactions between team members, are qualitative data.

Quantitative analysis uses numerical methods to ascertain the magnitude, amount, or size of something; for example, the attributes, behavior, or strength of opinion of the participants. For example, in describing a population, a quantitative analysis might conclude that the average person is 5 feet 11 inches tall, weighs 180 pounds, and is 45 years old.

Qualitative analysis focuses on the nature of something and can be represented by themes, patterns, and stories. For example, in describing the same population, a qualitative analysis might conclude that the average person is tall, thin, and middle-aged.

✓ First Steps in Analyzing Data

Having collected the data, some initial processing is normally required before data analysis can begin in earnest. For example, audio data may be transcribed by hand or by using an automated tool, such as Dragon; quantitative data, such as time taken or errors made, is usually entered into a spreadsheet, like Excel. Initial analysis steps for data typically collected through interviews, questionnaires, and observation are summarized in Table 6.1.

Table 6.1 Data gathered and typical initial processing steps for interviews, questionnaires, and observation

	Usual raw data	Example qualitative data	Example quantitative data	Initial processing steps
Interviews	Audio recordings. Interviewer notes. Video recordings.	Responses to open-ended questions. Video pictures. Respondent's opinions.	Age, job role, years of experience. Responses to close-ended questions.	Transcription of recordings. Expansion of notes. Entry of answers to close-ended questions into a spreadsheet
Questionnaires	Written responses. Online database.	Responses to open-ended questions. Responses in "further comments" fields. Respondent's opinions.	Age, job role, years of experience. Responses to close-ended questions.	Clean up data. Filter into different data sets. Synchronization between data recordings.
Observation	Observer's notes. Photographs. Audio and video recordings. Data logs. Think-aloud Diaries.	Records of behavior. Description of a task as it is undertaken. Copies of informal procedures.	Demographics of participants. Time spent on a task. The number of people involved in an activity. How many different types of activity are undertaken.	Expansion of notes. Transcription of recordings.

- **Interviews**

Interviewer notes need to be written up and expanded as soon as possible after the interview has taken place so that the interviewer's memory is clear and fresh. An audio or video recording may be used to help in this process, or it may be transcribed for more detailed analysis.

Transcription takes significant effort, as people talk more quickly than most people can type (or write), and the recording is not always clear. It is worth considering whether to transcribe the whole interview or just sections of it that are relevant. Deciding what is relevant, however, can be difficult. Revisiting the goals of the study to see which passages address the research questions can guide this process.

Close-ended questions are usually treated as quantitative data and analyzed using basic quantitative analysis. For example, a question that asks for the respondent’s age range can easily be analyzed to find out the percentage of respondents in each. More complicated statistical techniques are needed to identify relation- ships between responses that can be generalized, such as whether there is an interaction between the condition being tested and a demographic. Open-ended questions typically result in qualitative data that might be searched for categories or patterns of response.

- **Questionnaires**

Increasingly, questionnaire responses are provided using online surveys, and the data is automatically stored in a database. The data can be filtered according to respondent subpopulations (for instance, everyone under 16) or according to a particular question (for example, to understand respondents’ reactions to one kind of robot personality rather than another). This allows analyses to be conducted on subsets of the data and hence to draw specific conclusions for more targeted goals. To conduct this kind of analysis requires sufficient data from a large enough sample of participants.

- **Observation**

Observation can result in a wide variety of data including notes, photographs, data logs, think-aloud recordings (often called *protocols*), video, and audio recordings. Taken together, these different types of data can provide a rich picture of the observed activity. The difficult part is working out how to combine the different sources to create a coherent narrative of what has been recorded; analytic frameworks can help with this. Initial data processing includes writing up and expanding notes and transcribing elements of the audio and video recordings and the think-aloud protocols. For observation in a controlled environment, initial processing might also include synchronizing different data recordings.

Transcriptions and the observer’s notes are most likely to be analyzed using qualitative approaches, while photographs provide contextual information. Data logs and some elements of the observer’s notes would probably be analyzed quantitatively.

Lesson 3. Basic Analysis

- **Basic Quantitative Analysis (Sharp, 2019).**

Explaining statistical analysis requires a whole book on its own (for example. Here, we introduce two basic quantitative analysis techniques that can be used effectively in interaction design: averages and percentages. Percentages are useful for standardizing the data, particularly to compare two or more large sets of responses.

Averages and percentages are fairly well-known numerical measures. However, there are three different types of average, and using the wrong one can lead to the misinterpretation of the results. These three are: mean, median, and mode. *Mean* refers to the commonly understood interpretation of average; that is, add together all the figures and divide by the number of figures with which you started. Median and mode averages are less well-known but are very useful. The *median* is the middle value of the data when the numbers are ranked. The *mode* is the most commonly occurring number. For example, in a set of data (2, 3, 4, 6, 6, 7, 7, 7, 8), the median is 6 and the mode is 7, while the mean is 50/9 = 5.56. In this case, the difference between the different averages is not that great. However, consider the set (2, 2, 2, 2, 450). Now the median is 2, the mode is 2, and the mean is 458/5 = 91.6!

Use of simple averages can provide useful overview information, but they need to be used with caution. Evangelos Karapanos et al. (2009) go further and suggest that averaging treats diversity among participants as error and proposes the use of a multidimensional scaling approach instead.

Before any analysis can take place, the data needs to be collated into analyzable data sets. Quantitative data can usually be translated into rows and columns, where one row equals one record, such as respondent or interviewee. If these are entered into a spreadsheet such as Excel, this makes simple manipulations and data set filtering easier. Before entering data in this way, it is important to decide how to represent the different possible answers. For example, “don’t know” represents a different response from no answer at all, and they need to be distinguished, for instance, with separate columns in the spreadsheet. Also, if dealing with options from a close-ended question, such as job role, there are two different possible approaches that affect the analysis. One approach is to have a column headed “Job role” and to enter the job role as it is given by the respondent or interviewee. The alternative approach is to have a column for each possible answer. The latter approach lends itself more easily to automatic summaries. Note, however, that this option will be open only if the original question was designed to collect the appropriate data.

For simple collation and analysis, spreadsheet software such as Excel or Google Sheets is often used as it is commonly available, is well understood, and offers a variety of numerical manipulations and graphical representations. Basic analysis might involve finding out averages and identifying *outliers*, in other words, values that are significantly different from the majority, and hence not common. Producing a graphical representation provides an overall view of the data and any patterns it contains. Other tools are avail- able for performing specific statistical tests, such as online t-tests and A/B testing tools. Data visualization tools can create more sophisticated representations of the data such as heatmaps.

- **Basic Qualitative Analysis (Sharp, 2019).**

Three basic approaches to qualitative analysis are discussed in this section: identifying themes, categorizing data, and analyzing critical incidents. *Critical incident analysis* is a way to isolate subsets of data for more detailed analysis, perhaps by identifying themes or applying categories. These three basic approaches are not mutually exclusive and are often used in combination, for example, when analyzing video material critical incidents may first be identified and then a thematic analysis undertaken.

As with quantitative analysis, the first step in qualitative analysis is to gain an overall impression of the data and to start looking for interesting features, topics, repeated observations, or things that stands out. Some of these will have emerged during data gathering, and this may already have suggested the kinds of pattern to look for, but it is important to confirm and re-confirm findings to make sure that initial impressions don’t bias analysis. For example, you might notice from the logged data of people visiting TripAdviser.com that they often look for reviews for hotels that are rated “terrible” first. Or, you might notice that a lot of respondents all say how frustrating it is to have to answer so many security questions when logging onto an online banking service. During this first pass, it is not necessary to capture all of the findings but instead to highlight common features and record any surprises that arise (Blandford, 2017).

For observations, the guiding framework used in data gathering will give some structure to the data. For example, the practitioner’s framework for observation introduced in Chapter 8, “Data Gathering,” will have resulted in a focus on who, where, and what, while using the more detailed framework will result in patterns relating to physical objects, people’s goals, sequences of events, and so on.

Qualitative data can be analyzed inductively, that is, extracting concepts from the data, or deductively, in other words using existing theoretical or conceptual ideas to categorize data elements (Robson and McCartan, 2016). Which approach is used depends on the data obtained and the goal of the study, but the underlying principle is to classify elements of the data in order to gain insights toward the study's goal. Identifying themes (thematic analysis) takes an *inductive approach*, while categorizing data takes a *deductive approach*. In practice, analysis is often performed iteratively, and it is common for themes identified inductively then to be applied deductively to new data, and for an initial, pre-existing categorization scheme, to be enhanced inductively when applied to a new situation or new data. One of the most challenging aspects of identifying themes or new categories is determining meaningful codes that are orthogonal (that is, codes which do not overlap). Another is deciding on the appropriate granularity for them, for example at the word, phrase, sentence, or paragraph level. This is also dependent on the goal of the study and the data being analyzed.

Whether an inductive or deductive approach is used, an objective is to produce a reliable analysis, that is, one that can be replicated by someone else if they were to use the same type of approach. One way to achieve this is to train another person to do the coding. When training is complete, both researchers analyze a sample of the same data. If there is a large discrepancy between the two analyses, either training was inadequate or the categorization is not working and needs to be refined. When a high level of reliability is reached between the two researchers, it can be quantified by calculating the inter-rater reliability. This is the percentage of agreement between the analyses of the two researchers, defined as the number of items of agreement, for example the number of categories or themes arising from the data that have been identified consistently by both researchers, expressed as a percentage of the total number of items examined.

Lesson 4. Which Kind of Analytic Framework to Use? (Sharp, 2019).

There are several different analytical frameworks that can be used to analyze and interpret data from a qualitative study. In this section, six different approaches are outlined, ordered roughly in terms of their granularity, that is, the level of detail involved. For example, conversation analysis has a fine level of granularity, and it allows the details of what is said and how during a short fragment of conversation to be examined, while systems-based frameworks take a broader scope and have a coarser level of granularity, such as what happens when a new digital technology is introduced into an organization, like a hospital. Conversation analysis may result in insights related to users' interactions through a collaboration technology, while systems- based analyses may result in insights related to changes in work practices, worker satisfaction, improvements in workflow, impact on an office culture, and so on. Table 6.2 lists the six approaches in terms of the main types data, focus, expected outcomes, and level of granularity.

Table 6.2 Overview of analytical frameworks used in interaction Design

Framework	Data	Focus	Expected outcomes	Level of granularity
Conversation analysis	Recordings of spoken conversations	How conversations are conducted	Insights into how conversations are managed and how they progress	Word-level, or finer, for instance, pauses and inflection
Discourse analysis	Recordings of speech or writing from individuals or several participants	How words are used to convey meaning	Implicit or hidden meanings in texts	Word, phrase, or sentence-level
Content analysis	Any form of "text" including written pieces, video and audio recordings, or photographs	How often something is featured or is spoken about	Frequency of items appearing in a text	A wide range of levels from words, to feelings or attitudes, to artifacts or people
Interaction analysis	Video recordings of a naturally-occurring activity	Verbal and non-verbal interactions between people and artifacts	Insights about how knowledge and action are used within an activity	At the level of artifact, dialogue, and gesture
Grounded theory	Empirical data of any kind	Constructing a theory around the phenomenon of interest	A theory grounded in empirical data	Varying levels, depending on the phenomenon of interest
Systems-based frameworks	Large-scale and heterogeneous data	Large-scale involving people and technology, such as a hospital or airport	Insights about organizational effectiveness and efficiency	Macro-level, organizational level

Conversation Analysis

Conversation analysis (CA) examines the semantics of a conversation in fine detail. The focus is on how a conversation is conducted (Jupp, 2006). This technique is used in sociological studies, and it examines how conversations start and how turn-taking is structured, together with other rules of conversation. It has been used to analyze interactions in a range of settings, and it has influenced designers' understanding of users' needs in these environments. It can also be used to compare conversations that take place through different media, for example, face-to-face conversations versus those conducted through social media. More recently, it has been used to analyze the conversations that take place with voice-assisted technologies and chatbots.

Voice assistants (also called smart speakers), like Amazon Echo, have become increasingly popular in domestic settings, providing a limited kind of conversational interaction, mainly by answering questions and responding to requests.

Discourse Analysis

Discourse analysis focuses on dialogue, in other words, the meaning of what is said and how words are used to convey meaning. Discourse analysis is strongly interpretive, pays great attention to context, and views language not only as reflecting psychological and social aspects but also as constructing them (Coyle, 1995). An underlying assumption of discourse analysis is that there is no objective scientific truth. Language is a form of social reality that is open to interpretation from different perspectives. In this sense, the underlying philosophy of discourse analysis is similar to that of ethnography. Language is viewed as a constructive tool, and discourse analysis provides a way of focusing on how people use language to construct versions of their worlds (Fiske, 1994).

Small changes in wording can change meaning, as the following excerpts indicate:

> ***Discourse analysis is what you do when you are saying that you are doing discourse analysis. . .***
>
> ***According to Coyle, discourse analysis is what you do when you are saying that you are doing discourse analysis. . .***

By adding just three words, "According to Coyle," the sense of authority changes, depending on what the reader knows about Coyle's work and reputation.

Discourse analysis is useful when trying to identify subtle and implicit meaning in what people are writing about, what is trending, what is fake news, and so on. It can be used with data from interviews; in social media such as Facebook, Twitter, and WhatsApp; and in emails.

This kind of analysis of public discourse, when done by hand, is extremely time-consuming. To help, there are new software tools being developed that can automatically process computer- mediated discourses.

Content Analysis

Content analysis typically involves classifying the data into themes or categories and then studying the frequency of category occurrences (Krippendorff, 2013). The technique can be used for any text, where "text" refers to a range of media including video, newspapers, advertisements, survey responses, images, sounds, and so on. It can be used to analyze any online content, including the text of tweets, links, animated gifs, videos, and images.

Content analysis is often used in conjunction with other analysis techniques as well. They used content analysis alongside *sentiment analysis*, an approach that extracts emotional and subjective information from natural language. From their results, they found how feelings of sorrow and sadness were shared across the entire city because of the trauma associated with entrapment indoors during the deluge.

Interaction Analysis

Interaction analysis was developed by Brigitte Jordan and Austin Henderson (1995) as a way of investigating and understanding the interactions of human beings with each other and objects in their environment. The technique focuses on both talk and nonverbal inter- actions with artifacts and technologies, and it is based on video recordings. An underlying assumption of this approach is that knowledge and action are fundamentally social. The goal is to derive generalizations from videos of naturally occurring activities, focusing on how the people being observed make sense of each other's actions and their collective achievements.

Interaction analysis is an inductive process, where teams of researchers suggest statements about general patterns from multiple examples of empirical observations. Rather than individual researchers conducting separate analyses and then comparing their results for consistency, interaction analysis is conducted collaboratively; teams discuss together their observations and interpretations of the videos being watched as they watch them. The first step involves creating a content log, comprising headings and rough summaries of what has been observed. No predetermined categories are used during this stage. Instead, they emerge from repeated playing and discussion of the video material. Hypotheses are also generated by group members about what they think is happening. This process includes suggesting the intentions, motivations, and understandings of the people who are being viewed in the videos. These suggestions have to be tied to the actions of the people rather than being purely speculative. For example, if an analyst thinks someone's motivation is to take control during a board meeting, they need to provide actual examples that demonstrate how the person is achieving this (for instance, taking over the data projector, as the locus of control, for long periods of time and presenting their ideas for others to view).

Grounded Theory

The goal of *grounded theory* is to develop theory from a systematic analysis and interpretation of empirical data; that is, the derived theory is grounded in the data. In this respect, it is an inductive approach to developing theory.

Development of a "grounded" theory progresses through alternating data collection and data analysis: first data is collected and analyzed to identify themes, then that analysis may lead to further data collection and analysis to extend and refine the themes, and so on. During this cycle, parts of the data may be reanalyzed in more detail. Data gathering and subsequent analysis are hence driven by the emerging theory. This approach continues until no new insights emerge and the theory is well-developed. During this process, the researcher seeks to maintain a balance between objectivity and sensitivity. Objectivity is needed to maintain accurate and impartial interpretation of events; sensitivity is required to notice the subtleties in the data and identify relationships between concepts.

The thrust of the analysis undertaken is to identify and define the properties and dimensions of relevant themes called *categories* in grounded theory. According to Juliet Corbin and Anselm Strauss (2014), this coding has three aspects, which are iteratively performed through the cycle of data collection and analysis:

1. *Open coding* is the process through which categories, their properties, and dimensions are discovered in the data. This process is similar to our discussion of thematic analysis above, including the question of granularity of coding (at the word, line, sentence, conversation level, and so on).
2. *Axial coding* is the process of systematically fleshing out categories and relating them to their subcategories.

3. *Selective coding* is the process of refining and integrating categories to form a larger theoretical scheme. The categories are organized around one central category that forms the backbone of the theory. Initially, the theory will contain only an outline of the categories, but as more data is collected, they are refined and developed further.

The following analytic tools are used to help stimulate the analyst's thinking and identify and characterize relevant categories:

1. **The Use of Questioning:** In this context, this refers to questioning the data, not your participants. Questions can help an analyst to generate ideas or consider different ways of looking at the data. It can be useful to ask questions when analysis appears to be in a rut.
2. **Analysis of a Word, Phrase, or Sentence:** Considering in detail the meaning of an utterance can also help to trigger different perspectives on the data.
3. **Further Analysis Through Comparisons:** Comparisons may be made between objects or between abstract categories. In either case, comparing one with the other brings alternative interpretations.

Grounded theory uses thematic analysis that is, themes are identified from the data, but as data analysis informs data collection, it also relies on categorizing new data according to the existing thematic set and then evolving that set to accommodate new findings.

Lesson 5. Tools to Support Data Analysis (Sharp, 2019).

While it is possible to perform these kinds of data analysis using only manual techniques, most people would agree that it is quicker, easier, and more accurate to use a software tool of some kind in the majority of cases. Using a simple spreadsheet application is surprisingly effective, but there are other more sophisticated tools available to support the organization, coding, and manipulation of data, and to perform statistical tests.

Tools in the former category (to support the organization of data) include facilities for categorization, theme-based analysis, and quantitative analysis. These typically provide facilities to associate labels (categories, themes, and so on) with sections of data, search the data for key words or phrases, investigate the relationships between different themes or categories, and help to develop the coding scheme further. Some tools can also generate graphical representations. In addition, some provide help with techniques such as content analysis and sometimes mechanisms to show the occurrence and co-occurrence of words or phrases. In addition, searching, coding, project management, writing and annotating, and report generation facilities are common.

Two well-known tools that support some of these data analysis activities are ***Nvivo and Dedoose.*** For example, Nvivo supports the annotation and coding of data including PDF documents, photos, and video and audio files. Using Nvivo, field notes can be searched for key words or phrases to support coding or content analysis; codes and data can be explored, merged, and manipulated in several ways. The information can also be printed in a variety of forms such as a list of every occasion a word or phrase is used in the data, and a tree structure showing the relationships among codes. Like all software pack- ages, Nvivo has advantages and disadvantages, but it is particularly powerful for handling large sets of data and can generate output for statistical packages such as SAS and SPSS.

Statistical Analysis Software (SAS) and Statistical Package for the Social Sciences (SPSS) are popular quantitative analysis packages that support the use of statistical tests. SPSS, for example, is a sophisticated package offering a wide range of statistical tests such as frequency distributions, rank correlations (to determine statistical significance), regression analysis, and cluster analysis. SPSS assumes that the user knows and understands statistical analysis.

More information about software tools designed to support the analysis of qualitative data can be found through the CAQDAS Networking Project, based at the University of Surrey.
https://www.surrey.ac.uk/computer-assisted-qualitative-data-analysis

Lesson 6: Interpreting and Presenting the Findings (Sharp, 2019).

Previous sections in this chapter have illustrated a range of different ways to present findings—as tables of numbers and text, through various graphical devices and diagrams, as a set of themes or categories, and so on. Choosing an appropriate way to present the findings of a study is as important as choosing the right analytical approach. This choice will depend on the data gathering and analysis techniques used as well as the audience and the original goals of the study. In some situations, the details of data collection and analysis will be needed, for example, when working with others to make sense of a large collection of data, or when trying to convince an audience about a controversial conclusion. This detail may include snippets of data such as photographs of the context of use or videos of participants using the product. In other situations, only the salient trends, headlines, and overall implications are needed, so the style of presentation can be leaner. Where possible, a set of different complementary representations will be chosen to communicate the findings since any one representation will emphasize some aspects and de-emphasize others.

This section focuses on three kinds of presentation styles that haven't yet been emphasized to this point: using structured notations, stories, and summarizing.

- **Structured Notations**

A number of *structured notations* have been developed to analyze, capture, and present information for interaction design. These notations follow a clear syntax and semantics, which have been developed to capture particular viewpoints. Some are relatively straightforward, such as the work models promoted in contextual design (Beyer and Holtzblatt, 1998) that use simple conventions for representing flows, breakdowns, individual roles, and so on. Others, such as the modeling language Unified Modeling Language (UML), have stricter and more precise syntax to be followed and are often used to represent requirements; the activity diagrams, for example, are very expressive when detailed interactions need to be captured.

Advantages of using a structured notation are that the meaning of different symbols is well-defined, and so it provides clear guidance on what to look for in the data and what to highlight and that it enforces precision in expression. Disadvantages include that by high- lighting specific elements, it inevitably de-emphasizes or ignores other aspects, and the precision expressed by the notation may be lost on an audience if they don't know the notation well. Producing diagrams or expressions in these notations may require further analysis of the findings in order to identify the specific characteristics and properties that the notation highlights. To overcome these disadvantages, structured notations are usually used in combination with stories or other easily accessible formats.

- **Using Stories**

Storytelling is an easy and intuitive approach for people to communicate ideas and experiences. It is not surprising then that stories (also called *narratives*) are used extensively in interaction design, both to communicate findings of investigative studies and as the basis for further development, such as product design or system enhancements.

Storytelling may be employed in three different ways. First, participants (such as interviewees, questionnaire respondents, and those you have observed) may have told stories of their own during data gathering. These stories can be extracted, can be compared, and may be used to communicate findings to others, for example, to illustrate points.

Second, stories (or narratives) based on observation, such as ethnographic field studies, may be employed for further data gathering. The scenarios were developed on the basis of ethnographic studies and previous co-design activities and were presented through storytelling to facilitate understanding. Note that, in this case, the audience was a group of participants in the ongoing study.

Including specific stories gives authenticity to the findings, and it can add to its credibility provided the conclusions are not overstated. Making a multimedia presentation of the story by adding video or audio excerpts and photographs will illustrate the story further. This kind of approach can also be effective if presenting data from an evaluation study that involves observation, as it is hard to contest well-chosen video excerpts of users interacting with technology or extracts from interview transcripts.

Third, stories may be constructed from smaller snippets or repeated episodes that are found in the data. In this case, stories provide a way of rationalizing and collating data to form a representative account of a product's use or a certain type of event.

Any stories collected through data gathering may be used as the basis for constructing scenarios that can then be used for requirements and design activities.

- **Summarizing the Findings**

Presentation styles will usually be used in combination to produce a summary of the findings; for instance, a story may be expanded with graphical representations of activity or demo- graphics, and data excerpts from transcripts or videos may be used to illustrate particular points. Tables of numerical data may be represented as graphs, diagrams, or rigorous notations, together with workflows or quotations.

Careful interpretation and presentation of the study results is just as important as choosing the right analysis technique so that findings are not over-emphasized, and evidence is not misrepresented. Over-generalizing results without good evidence is a common pitfall, especially with qualitative analyses' for example, think carefully before using words such as *most*, *all*, *majority*, and *none*, and be sure that the justifications reflect the data. As discussed even statistical results can be interpreted in a misleading way. For example, if 8 out of 10 users preferred design A over design B, this does not mean that design A is 80 percent more attractive than design B. If you found 800 out of 1,000 users preferred design A, then you have more evidence to suggest that design A is better, but there are still other factors to consider.

Assessment Task

Activity No. 1

The aim of this activity is to practice data gathering. Go back to your activity given in Module 1 (eCommerce Web page). This existing project may be redesigned, or a completely new web design may be created. To do the assignment, find a group of people or a single individual prepared to be the user group. These could be your family, friends, peers, or people in a local community group.

For this assignment:

(a) Clarify the basic goal of improving the product by considering what this means in your considerations.

(b) Watch the group (or person) casually to get an understanding of any issues that might create challenges for this activity and any information to help refine the study goals.

(c) Explain how you would use each of the three data gathering techniques: interview, questionnaire, and observation in your data gathering program. Explain how your plan takes account of triangulation.

(d) Plan your data gathering program in detail.

- Decide what kind of interview to run and design a set of interview questions. Decide how to record the data, then acquire and test any equipment needed and run a pilot study.
- Decide whether to include a questionnaire in your data gathering program, and design appropriate questions for it. Run a pilot study to check the questionnaire.

- Decide whether to use direct or indirect observation and where on the outsider/insider spectrum should the observers be. Decide how to record the data, then acquire and test any equipment needed and run a pilot study.

(e) Carry out the study, but limit its scope. For example, interview only two or three people or plan only two half-hour observation periods.

(f) Reflect on this experience and suggest what you would do differently next time.

Activity No. 2

The goal of this in-depth activity is to practice data analysis and presentation. Refer to your data gathering done during the previous module activity, you are assigned to analyze and present the findings of your data gathering activity from the previous module to a group of peers, in your class.

1. Review the data that you gathered and identify any qualitative data and any quantitative data in the data set.
2. Is there any qualitative data that could sensibly and helpfully be translated into quantitative measures? If so, do the translation and add this data to your quantitative set.
3. Consider your quantitative data.
   (a) Decide how best to enter it into spreadsheet software, for example, how to handle answers to close-ended questions. Then enter the data and generate some graphical representations. As the data set is likely to be small, think carefully about what, if any, graphical representations will provide meaningful summaries of the findings.
   (b) Is there any data for which simple measures, such as percentages or averages, will be helpful? If so, calculate the three different types of average.
4. Consider your qualitative data.
   (a) Based on your refinement of the study question "improving the product," identify some themes in the qualitative data, for example, what features of the product cause people difficulties? Did any of the participants suggest alternative designs or solutions? Refine your themes and collate extracts of data that support the theme.
   (b) Identify any critical incidents in the data. These may arise from interviews, questionnaire responses, or observation. Describe these incidents carefully and choose one or two to

analyze in more depth, focusing on the context in which they occurred.

5. Collate your findings as a presentation and present them to a group of peers.
6. Review the presentation and any questions from the audience. Consider how to improve the analysis and presentation.

Summary

This module has focused on three main data gathering methods that are commonly used in interaction design: interviews, questionnaires, and observation. It has described in detail the planning and execution of each. In addition, five key issues of data gathering were presented, and how to record the data gathered was discussed.

Points to Remember

- All data gathering sessions should have clear goals.
- Depending on the study context, an informed consent form and other permissions may be needed to run the study.
- Running a pilot study helps to test out the feasibility of a planned data gathering session and associated instruments such as questions.
- Triangulation involves investigating a phenomenon from different perspectives.
- Data may be recorded using handwritten notes, audio or video recording, a camera, or any combination of these.
- There are three styles of interviews: structured, semi-structured, and unstructured.
- Questionnaires may be paper-based, via email, or online.
- Questions for an interview or questionnaire can be open or closed-ended. Closed-ended questions require the interviewee to select from a limited range of options. Open-ended questions accept a free-range response.
- Observation may be direct or indirect.
- In direct observation, the observer may adopt different levels of participation, ranging from insider (participant observer) to outsider (passive observer).
- Choosing appropriate data gathering techniques depends on the focus of the study, participants involved, nature of the technique, and resources available.

Also, this module described in detail the difference between qualitative and quantitative data and between qualitative and quantitative analysis.

Quantitative and qualitative data can be analyzed for patterns and trends using simple techniques and graphical representations. Qualitative data may be analyzed inductively or deductively using a variety of approaches. Thematic analysis (an example of inductive analysis) and data categorization (an example of deductive analysis) are common approaches. Analytical frameworks include conversation analysis, discourse analysis, content analysis, interaction analysis, grounded theory, and systems-based approaches.

It was noted that presenting the results is just as important as analyzing the data, hence it is important to make sure that any summary or claim arising from the analysis is carefully contextualized, and that it can be justified by the data.

Points to Remember

- The kind of data analysis that can be done depends on the data gathering techniques used.
- Qualitative and quantitative data may be collected from any of the main data gathering techniques: interviews, questionnaires, and observation.
- Quantitative data analysis for interaction design usually involves calculating percentages and averages.
- There are three different kinds of average: mean, mode, and median.
- Graphical representations of quantitative data help in identifying patterns, outliers, and the overall view of the data.
- Analysis of qualitative data analysis may be inductive, in which themes or categories are extracted from the data, or deductive, in which pre-existing concepts are used to interrogate the data.
- In practice, analysis often proceeds in iterative cycles combining inductive identification of themes and deductive application of categories and new themes.
- Which analysis approach is used is tightly coupled to the data that is collected and depends on the goals of the study.

- Several analytical frameworks exist that focus on different levels of granularity with different purposes.

References

a. Karapanos, E., Martensi, J.-B., and Hassenzahl, M. (2009) Accounting for Diversity in Subjective Judgments. In *Proceedings of CHI 2009*. ACM, New York, NY,

b. Blandford, A., Furniss, D., and Makri, S. (2017) *Qualitative HCI Research: Going Behind the Scenes*. Morgan Claypool Publishers.

c. Jupp, V. (ed.) (2006) *The SAGE Dictionary of Social Research Methods*. SAGE.

d. Coyle, A. (1995) Discourse Analysis. In G. M. Breakwell, S. Hammond and C. Fife-Schaw (eds) *Research Methods in Psychology*. SAGE, London.

e. Corbin, J. M., and Strauss, A. (2014) *Basics of Qualitative Research: Techniques and Proce- dures for Developing Grounded Theory*. SAGE Publications.

f. Sharp, H., Preece, J., and Rogers Y. (2019) *Interaction Design: Beyond Human-Computer Interaction - PDF Drive*. (2019). Pdfdrive.com. https://www.pdfdrive.com/interaction-design-beyond-human-computer-interaction-e190044261.html

MODULE 7
DISCOVERING REQUIREMENTS, DESIGN, PROTOTYPING AND CONSTRUCTION

Introduction

Discovering requirements focuses on exploring the problem space and defining what will be developed. In the case of interaction design, this includes: understanding the target users and their capabilities; how a new product might support users in their daily lives; users' current tasks, goals, and contexts; constraints on the product's performance; and so on. This understanding forms the basis of the product's requirements and underpins design and construction.

It may seem artificial to distinguish between requirements, design, and evaluation activities because they are so closely related, especially in an iterative development cycle like the one used for interaction design. In practice, they are all intertwined, with some design taking place while requirements are being discovered and the design evolving through a series of evaluation—redesign cycles. With short, iterative development cycles, it's easy to confuse the purpose of different activities. However, each of them has a different emphasis and specific goals, and each of them is necessary to produce a quality product.

This module describes the requirements activity in more detail, and it introduces some techniques specifically used to explore the problem space, define what to build, and characterize the target audience.

Design, prototyping, and construction fall within the Develop phase of the double diamond of design, introduced in the previous module, "The Process of Interaction Design," in which solutions or concepts are created, prototyped, tested, and iterated. The final product emerges iteratively through repeated design-evaluation-redesign cycles involving users, and prototypes facilitate this process. There are two aspects to design: the conceptual part, which focuses on the idea of a product, and the concrete aspect, which focuses on the details of the design. The former involves developing a conceptual model that captures what the product will do and how it will behave, while the latter is concerned with the details of the design, such as menu types, haptic feedback, physical widgets, and graphics. The two are intertwined, and concrete design issues will require some consideration in order to prototype ideas, and prototyping ideas will lead to an evolution of the concept.

For users to evaluate the design of an interactive product effectively, designers prototype their ideas. In the early stages of development, these prototypes may be made of paper and cardboard, or ready-made components pulled together to allow evaluation, while as the design progresses, they become more polished, tailored, and robust so that they resemble the final product.

Learning Outcomes

At the end of this module, students should be able to:

1. Describe different kinds of requirements.
2. Allow you to identify different kinds of requirements from a simple description.
3. Explain additional data gathering techniques and how they may be used to discover requirements.
4. Enable you to develop a persona and a scenario from a simple description.
5. Describe use cases as a way to capture interaction in detail.
6. Describe prototyping and the different types of prototyping activities.
7. Enable you to produce simple prototypes from the models developed during the requirements activity.
8. Enable you to produce a conceptual model for a product and justify your choices.
9. Explain the use of scenarios and prototypes in design.
10. Introduce both physical computing kits and software development kits and their role in construction.

Lesson 1. What, How and Why? (Sharp, 2019)

- **What Is the Purpose of the Requirements Activity?**

The *requirements activity* sits in the first two phases of the double diamond of design, introduced in the previous modules, “The Process of Interaction Design.” These two phases involve exploring the problem space to gain insights about the problem and establishing a description of what will be developed. The techniques described in this chapter support these activities, and they capture the outcomes in terms of requirements for the product plus any supporting artifacts.

Requirements may be discovered through targeted activities, or tangentially during product evaluation, prototyping, design, and construction. Along with the wider interaction design lifecycle, requirements discovery is iterative, and the iterative cycles ensure that the les- sons learned from any of these activities feed into each other. In practice, requirements evolve and develop as the stakeholders interact with designs and learn what is possible and how features can be used. And, as shown in the interaction design lifecycle model in previous module, the activity itself will be repeatedly revisited.

➢ How to Capture Requirements Once They Are Discovered?

Requirements may be captured in several different forms. For some products, such as an exercise monitoring app, it may be appropriate to capture requirements implicitly through a prototype or operational product. For others, such as process control software in a fac- tory, a more detailed understanding of the required behavior is needed before prototyping or construction begins, and a structured or rigorous notation may be used to investigate the product's requirements. In all cases, capturing requirements explicitly is beneficial in order to make sure that key requirements aren't lost through the iterations. Interactive products span a wide range of domains with differing constraints and user expectations. Although it may be disappointing if a new app to alert shoppers about offers on their favorite purchases turns out to be unusable or slightly inaccurate, if the same happens to an air traffic control system, the consequences are far more significant and could threaten lives.

There are different kinds of requirements, and each can be emphasized or de-emphasized by different notations because notations emphasize different characteristics. For example, requirements for a product that relies on processing large amounts of data will be captured using a notation that emphasizes data characteristics. This means that a range of representations is used including prototypes, stories, diagrams, and photographs, as appropriate for the product under development.

• What Are Requirements?

A *requirement* is a statement about an intended product that specifies what it is expected to do or how it will perform. For example, a requirement for a smartwatch GPS app might be that the time to load a map is less than half a second. Another, less precise requirement might be for teenagers to find the smartwatch appealing, and the requirements activity would involve exploring in more detail exactly what would make such a watch appealing to teenagers.

One of the goals of the requirements activity is to identify, clarify, and capture the requirements. The process of discovering requirements is iterative, allowing requirements and their understanding to evolve. In addition to capturing the requirements themselves, this activity also involves specifying criteria that can be used to show when the requirements have been fulfilled. For example, usability and user experience criteria can be used in this way.

Requirements come in different forms and at different levels of abstraction. An alternative way to capture what a product is intended to do is via user stories. *User stories* communicate requirements between team members. Each one represents a unit of customer-visible functionality and serves as a starting point for a conversation to extend and clarify requirements. User stories may also be used to capture usability and user experience goals. Originally, user stories were normally written on physical cards that deliberately con- strained the amount of information that could be captured in order to prompt conversations between stakeholders. While these conversations are still highly valued, the use of digital support tools such as Jira *(https://www.atlassian.com/software/jira)* has meant that additional information to elaborate the requirement is often stored with user stories.

User stories are most prevalent when using an agile approach to product development. User stories form the basis of planning for a sprint and are the building blocks from which the product is constructed. Once completed and ready for development, a story consists of a description, an estimate of the time it will take to develop, and an acceptance test that determines how to measure when the requirement has been fulfilled. It is common for a user story such as the earlier ones to be decomposed further into smaller stories, often called *tasks*.

During the early stages of development, requirements may emerge in the form of epics. An *epic* is a user story that may take weeks or months to implement. Epics will be broken down into smaller chunks of effort (user stories), before they are pulled into a sprint. Example epics for a travel organizer app might be the following:

- As a *<group traveler>*, I want *<to choose from a range of potential vacations that suit the group's preferences>* so that *<the whole group can have a good time>*.
- As a *<group traveler>*, I want *<to know the visa restrictions for everyone in the group>*

so that *<visas can be arranged for everyone in the group in plenty of time>*.

- As a *<group traveler>*, I want *<to know the vaccinations required to visit the chosen destination>* so that *<vaccinations can be arranged for everyone in the group in plenty of time>*.
- As a *<travel agent>*, I want *<up-to-date information displayed>* so that *<my clients receive accurate information>*.

Different Kinds of Requirements

Requirements come from several sources: from the user community, from the business com- munity, or as a result of the technology to be applied. Two different kinds of requirements have traditionally been identified: ***functional requirements***, which describe what the product will do, and nonfunctional requirements, which describe the characteristics (sometimes called *constraints*) of the product. For example, a functional requirement for a new video game might be that it will be challenging for a range of user abilities. This requirement might then be decomposed into more specific requirements detailing the structure of challenges in the game, for instance, levels of mastery, hidden tips and tricks, magical objects, and so on. A ***nonfunctional requirement*** for this same game might be that it can run on a variety of platforms, such as the Microsoft Xbox, Sony PlayStation, and Nintendo Switch game systems. Interaction design involves understanding both functional and nonfunctional requirements.

There are many more different types of requirements, however. Suzanne and James Robertson (2013) suggest a comprehensive categorized set of requirements types (see Table 7.1), while Ellen Gottesdiener and Mary Gorman (2012) suggest seven product dimensions (see Figure 7.1).

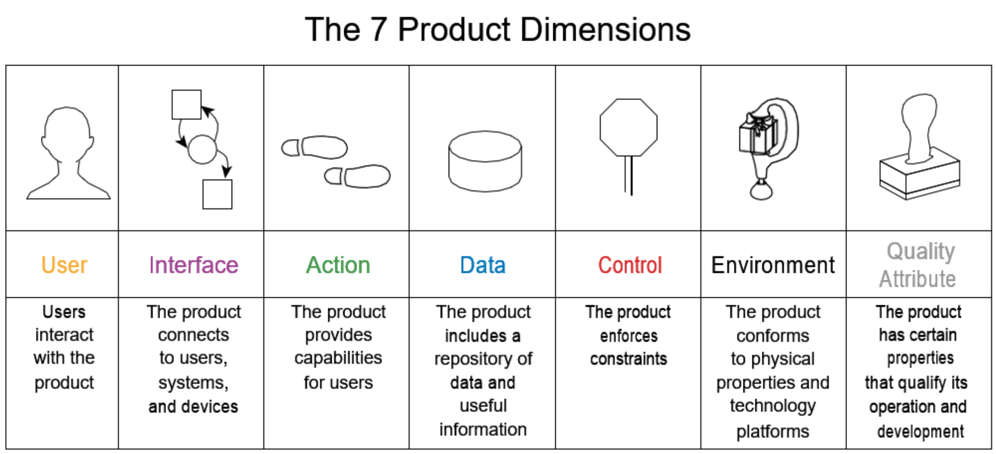
Table 7.1 A comprehensive categorization of requirements types

Source: Atlantic Systems Guild, Volere Requirements Specification Template, Edition 18 (2017), http://www.volere.co.uk/template.htm

Project Drivers	1. The Purpose of the Product
	2. The Stakeholders
Project Constraints	3. Mandated Constraints
	4. Naming Conventions and Terminology
	5. Relevant Facts and Assumptions
Functional Requirements	6. The Scope of the Work
	7. Business Data Model and Data Dictionary
	8. The Scope of the Product
	9. Functional Requirements

Nonfunctional Requirements	10. Look and Feel Requirements
	11. Usability and Humanity Requirements
	12. Performance Requirements
	13. Operational and Environmental Requirements
	14. Maintainability and Support Requirements
	15. Security Requirements
	16. Cultural Requirements
	17. Compliance Requirements
Project Issues	18. Open Issues
	19. Off-the-Shelf Solutions
	20. New Problems
	21. Tasks
	22. Migration to the New Product
	23. Risks
	24. Costs
	25. User Documentation and Training
	26. Waiting Room
	27. Ideas for Solutions

Figure 7.1 The seven product dimensions



Source: Gottesdiener and Gorman (2012), p. 58. Used courtesy of Ellen Gottesdiener

> To see how the seven product dimensions can be used to discover requirements, see **https://www.youtube.com/watch?v=x9olpZaXTDs**.

In this section, six of the most common types of requirements are discussed: functional, data, environment, user, usability, and user experience.

Functional requirements capture what the product will do. For example, a functional requirement for a robot working in a car assembly plant might be that it is able to place and weld together the correct pieces of metal accurately. Understanding the functional requirements for an interactive product is fundamental.

Data requirements capture the type, volatility, size/amount, persistence, accuracy, and value of the required data. All interactive products have to handle some data. For example, if an application for buying and selling stocks and shares is being developed, then the data must be up-to-date and accurate, and it is likely to change many times a day. In the personal banking domain, data must be accurate and persist over many months and probably years, and there will be plenty of it.

Environmental requirements, or context of use, refer to the circumstances in which the interactive product will operate. Four aspects of the environment lead to different types of requirements. First is the ***physical environment***, such as how much lighting, noise, movement, and dust is expected in the operational environment. Will users need to wear protective clothing, such as large gloves or headgear that might affect the choice of interface type? How crowded is the environment? For example, an ATM operates in a very public physical environment, thus using a speech interface is likely to be problematic.

The second aspect of the environment is the ***social environment***. Issues regarding the social aspects of interaction design, such as collaboration and coordination. For example, will data need to be shared? If so, does the sharing have to be synchronous (for instance, viewing the data at once) or asynchronous (for example, two people authoring a report taking turns to edit it)? Other factors include the physical location of fellow team members, such as collaborators communicating across great distances.

The third aspect is the ***organizational environment***, for example, how good is user sup- port likely to be, how easily can it be obtained, and are there facilities or resources for training, how efficient or stable is the communications infrastructure, and so on?

Finally, the ***technical environment*** will need to be established. For example, what technologies will the product run on or need to be compatible with, and what technological limitations might be relevant?

User characteristics capture the key attributes of the intended user group, such as the users' abilities and skills, and depending on the product, also their educational background, preferences, personal circumstances, physical or mental disabilities, and so on. In addition, a user may be a novice, an expert, a casual user, or a frequent user. This affects the ways in which interaction is designed. For example, a novice user may prefer step-by-step guidance. An expert, on the other hand, may prefer a flexible interaction with more wide-ranging pow- ers of control. The collection of characteristics for a typical user is called a *user profile.* Any one product may have several different user profiles.

Usability goals and *user experience goals* are another kind of requirement, and they should be captured together with appropriate measures. Previously *usability engineering* briefly introduced, it was an approach in which specific measures for the usability goals of the product are agreed upon early in the development process and are used to track progress as development proceeds. This both ensures that usability is given due priority and facilitates progress tracking. The same is true for user experience goals. Although it is harder to identify quantifiable measures that track these qualities, an understanding of their importance is needed during the requirements activity.

Different interactive products will be associated with different requirements. For example, a telecare system designed to monitor an elderly person's movements and alert relevant care staff will be constrained by the type and size of sensors that can be easily worn by the users as they go about their normal activities. Wearable interfaces need to be light, small, fashionable, preferably hidden, and not get in the way. A desirable characteristic of both an online shopping site and a robotic companion is that they are trustworthy, but this attribute leads to different nonfunctional requirements—in the former, security of information would be a priority, while in the latter behavioral norms would indicate trustworthiness. A key requirement in many systems nowadays is that they be secure, but one of the challenges is to provide security that does not detract from the user experience.

Lesson 2. Prototyping (Sharp, 2019).

It is often said that users can't tell you what they want, but when they see something and get to use it, they soon know what they don't want. *Prototyping* provides a concrete manifestation of an idea—whether it is a new product or a modification of an existing one—which allows designers to communicate their ideas and users to try them out.

- **What Is a Prototype?**

  A *prototype* is one manifestation of a design that allows stakeholders to interact with it and to explore its suitability. It is limited in that a prototype will usually emphasize one set of product characteristics and de-emphasize others. Prototypes take many forms, for example, a scale model of a building or a bridge, or a piece of software that crashes every few minutes. A prototype can also be a paper-based outline of a display, a collection of wires and ready-made components, a digital picture, a video simulation, a complex piece of soft- ware and hardware, or a three-dimensional mockup of a workstation.

  In fact, a prototype can be anything from a paper-based storyboard to a complex piece of software, and from a cardboard mockup to a molded or pressed piece of metal.

- **Why Prototype?**

  Prototypes are useful when discussing or evaluating ideas with stakeholders; they are a communication device among team members and an effective way for designers to explore design ideas. The activity of building prototypes encourages reflection in design, and it is recognized by designers from many disciplines as an important aspect of design.

  Prototypes answer questions and support designers in choosing between alternatives. Hence, they serve a variety of purposes, for example, to test the technical feasibility of an idea, to clarify some vague requirements, to do some user testing and evaluation, or to check that a certain design direction is compatible with the rest of product development. The purpose of a prototype will influence the kind of prototype that is appropriate to build. So, for example, to clarify how users might perform a set of tasks and whether the proposed design would support them in doing this, a paper-based mockup might be produced. Figure 7.2 shows a paper-based prototype of a handheld device to help an autistic child communicate. This prototype shows the intended functions and buttons, their positioning and labeling, and the overall shape of the device, but none of the buttons actually works. This kind of prototype is sufficient to investigate scenarios of use and to decide, for example, whether the button images and labels are appropriate and the functions sufficient, but not to test whether the speech is loud enough or the response fast enough.

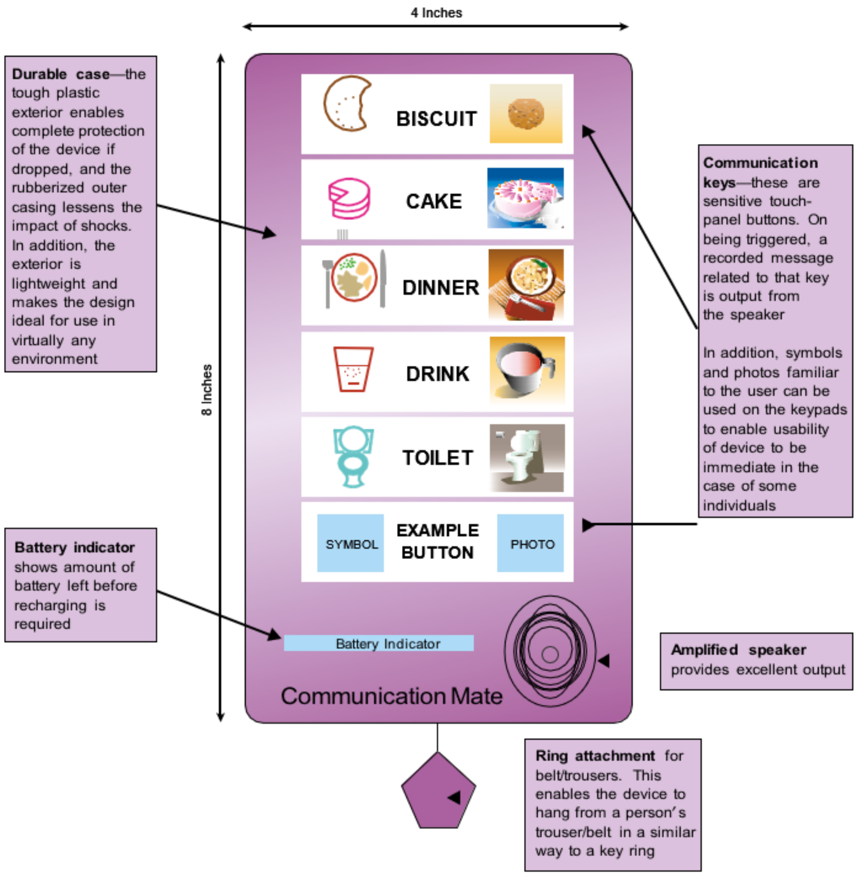


Figure 7.2 A paper-based prototype of a handheld device to support an autistic child
Source: Used courtesy of Sigil Khwaja

> To see more about the development of the Wynd Halo and Wynd Home Purifier, see this website:
> **https://www.kickstarter.com/projects/882633450/wynd-halo-home-purifier-keep-your-homes-air-health?ref=discovery**

Low-Fidelity Prototyping

A *low-fidelity prototype* does not look very much like the final product, nor does it provide the same functionality. For example, it may use very different materials, such as paper and cardboard rather than electronic screens and metal, it may perform only a limited set of functions, or it may only represent the functions and not perform any of them.

Low-fidelity prototypes are useful because they tend to be simple, cheap, and quick to produce. This also means that they are simple, cheap, and quick to modify so that they sup- port the exploration of alternative designs and ideas. This is particularly important in the early stages of development, during conceptual design for example, because prototypes that are used for exploring ideas should be flexible and encourage exploration and modification. Low-fidelity prototypes are not meant to be kept and integrated into the final product.

Storyboarding

Storyboarding is one example of low-fidelity prototyping that is often used in conjunction with scenarios. A *storyboard* consists of a series of sketches showing how a user might progress through a task using the product under development. It can be a series of screen sketches or a series of scenes showing how a user can perform a task using an interactive device. When used in conjunction with a scenario, the storyboard provides more detail and offers stakeholders a chance to role-play with a prototype, interacting with it by stepping through the scenario. The example storyboard shown in Figure 7.3 depicts a person using a new mobile device for exploring historical sites. This example shows the context of use for this device and how it might support her quest for information about the pottery trade at The Acropolis in Ancient Greece.

Figure 7.3 An example storyboard for a mobile device to explore ancient sites such as The Acropolis

Sketching

Low-fidelity prototyping often relies on hand-drawn sketches. Many people find it difficult to engage in *sketching* because they are inhibited by the quality of their drawing. As Saul Greenberg et al. (2012) put it, however, "Sketching is not about drawing. You can get over any inhibition by devising your own symbols and icons and practicing them—referred to by Saul Greenberg et al. as a *sketching vocabulary*. They don’t have to be anything more than simple boxes, stick figures, and stars. Elements that might be required in a storyboard sketch, for example, include digital devices, people, emotions, tables, books, and so forth, and actions such as give, find, transfer, and write. If you are sketching an interface design, then you might need to draw various icons, dialog boxes, and so on. Some simple examples are shown in Figure 7.4.

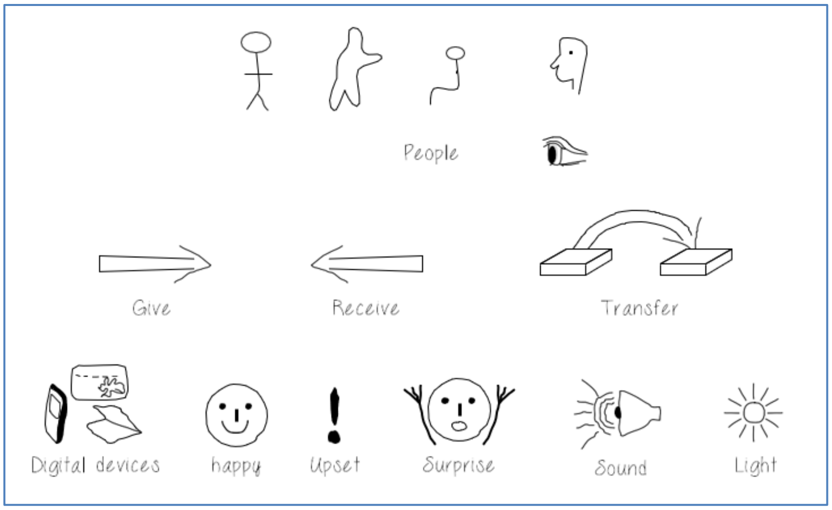


Figure 7.4 Some simple sketches for low-fidelity prototyping

Prototyping with Index Cards

Using *index cards* (small pieces of cardboard about 3 × 5 inches) is a successful and simple way to prototype an interaction, and it is used for developing a range of interactive products including websites and smartphone apps. Each card represents one element of the interaction, perhaps a screen or just an icon, menu, or dialog exchange. In user evaluations, the user can step through the cards, pretending to perform the task while interacting with the cards.

Wizard of Oz

Another low-fidelity prototyping method called *Wizard of Oz* assumes that you have a software- based prototype. With this technique, the user interacts with the software as though interacting with the product. In fact, however, a human operator simulates the software’s response to the user. The method takes its name from the classic story of the little girl who is swept away in a storm and finds herself in the Land of Oz (Baum and Denslow, 1900). The Wizard of Oz is a small shy man who operates a large artificial image of himself from behind a screen where no one can see him. The Wizard of Oz technique is often used in human-robot interaction studies. One such example is *Marionette*, a Wizard of Oz system for performing studies on the road with autonomous vehicles (Wang et al., 2017). Prototyping AI systems also draws on this style of prototyping, where the designer sketches the AI for themselves, and as the design matures, implementations of the AI can take its place (van Allen, 2018).

High-Fidelity Prototyping

A *high-fidelity prototype* looks more like the final product and usually provides more functionality than a low-fidelity prototype. For example, a prototype of a software system developed in Python or other executable language is higher fidelity than a paper-based mock-up; a molded piece of plastic with a dummy keyboard would be a higher-fidelity prototype of the PalmPilot than the lump of wood. There is a continuum between low- and high-fidelity, and prototypes used "in the wild," for example, will have enough fidelity to be able to answer their design questions and to learn about interaction or technological constraints or contextual factors. It is common for prototypes to evolve through various stages of fidelity, within the design-evaluate-redesign cycles.

High-fidelity prototypes can be developed by modifying and integrating existing components—both hardware and software—which are widely available through various developer kits and open source software, for example. In robotics, this approach has been called *tinkering* (Hendriks-Jansen, 1996), while in software development it has been referred to as *Opportunistic System Development* (Ncube et al, 2008). For example, Ali Al-Humairi et al. (2018) used existing hardware (Arduino) and open-source software to build a prototype to test their idea of robotically playing musical instruments automatically from a mobile phone.

Lesson 3. Construction (Sharp, 2019).

As prototyping and building alternatives progresses, development will focus more on putting together components and developing the final product. This may take the form of a physical product, such as a set of alarms, sensors, and lights, a piece of software, or both. Whatever the final form, it is unlikely that anything will need to be developed from scratch, as there are many useful (in some cases essential) resources to support development. Here we introduce two kinds of resources: physical computing kits and software development kits (SDKs).

- **Physical Computing**

Physical computing is concerned with how to build and code prototypes and devices using electronics. Specifically, it is the activity of "creating physical artifacts and giving them behaviors through a combination of building with physical materials, computer programming, and circuit building" (Gubbels and Froehlich, 2014). Typically, it involves designing things, using a printed circuit board (PCB), sensors (for instance push buttons, accelerometers, infrared, or temperature sensors) to detect states, and output devices (such as displays, motors, or buzzers) that cause some effect. An example is a "friend or foe" cat detector that senses, via an accelerometer, any cat (or anything else for that matter) that tries to push through a family's cat door. The movement triggers an actuator to take a photo of what came through the cat door using a webcam positioned on the back door. The photo is uploaded to a website that alerts the owner if the image does not match that of their own cat.

A number of physical computing toolkits have been developed for educational and prototyping purposes. One of the earliest is Arduino (Banzi, 2009). The goal was to enable artists and designers to learn how to make and code physical prototypes using electronics in a couple of days, having attended a workshop. The toolkit is composed of two parts: the Arduino board (see Figure 7.5), which is the piece of hardware that is used to build objects, and the Arduino integrated development environment (IDE), which is a piece of software that makes it easy to program and upload a sketch (Arduino's name for a unit of code) to the board. A sketch, for example, might turn on an LED when a sensor detects a change in the light level. The Arduino board is a small circuit that contains a tiny chip (the microcontroller). It has two rows of small electrical "sockets" that let the user connect sensors and actuators to its input and output pins. Sketches are written in the IDE using a simple processing language and then translated into the C programming language and uploaded to the board.

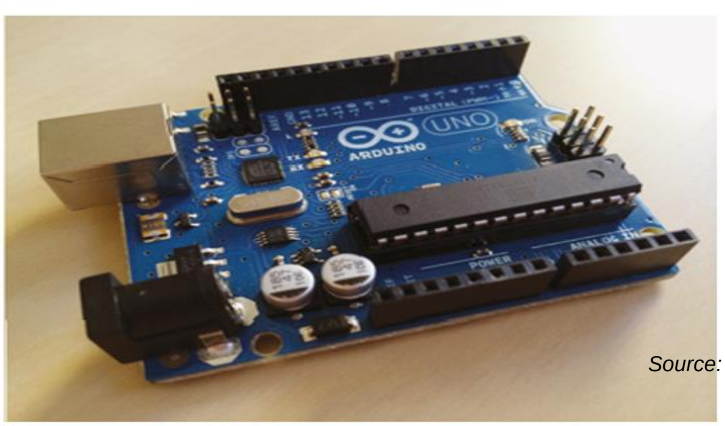


Figure 7.5 The Arduino board
Source: Used courtesy of Dr Nicolai Marquardt

One of the most recent physical computing systems is the BBC micro:bit (see Figure 7.5). Like Arduino, the micro:bit system consists of a physical computing device that is used in conjunction with an IDE. However, unlike Arduino, the micro:bit device contains a number of built-in sensors and a small display so that it is possible to create simple physical computing systems without attaching any components or wires. If desired, external components can still be added, but rather than the small electrical sockets of the Arduino, the micro:bit has an "edge connector" for this purpose. This is formed from a row of connection points that run along one edge of the device and allow it to be "plugged into" a range of accessories including larger displays, Xbox-style game controllers, and small robots. The micro:bit IDE, which runs in a web browser with no installation or setup process, supports a graphical programming experience based on visual "blocks" of code alongside text-based editing using a variant of JavaScript. This means that the micro:bit provides a great experience for young students and other beginner programmers, while also supporting more sophisticated programming. As a result, micro:bit has been widely adopted in schools around the world.

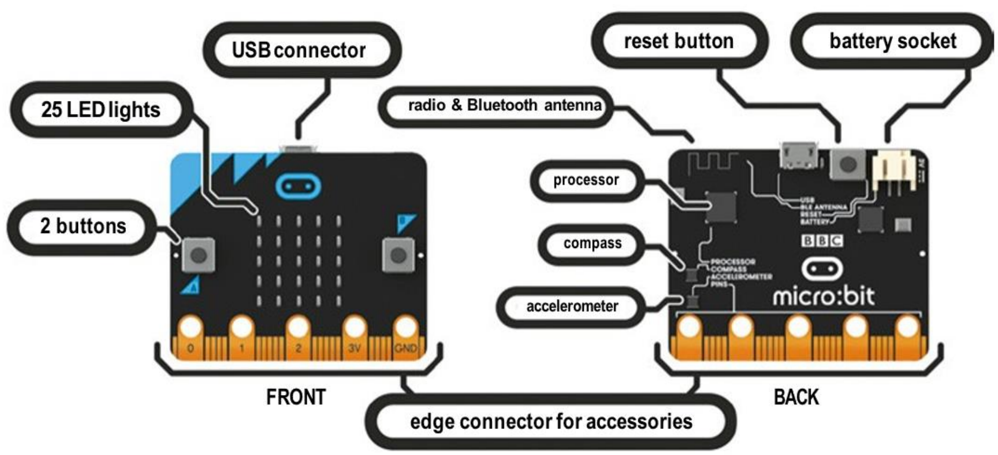


Figure 7.6 The BBC micro:bit

Source: https://microbit.org/guide/features. Used courtesy of Micro:bit Foundation

SDKs: Software Development Kits

A *software development kit* (SDK) is a package of programming tools and components that supports the development of applications for a specific platform, for example, for iOS on iPhone and iPad and for Android on mobile phone and tablet apps. Typically, an SDK includes an integrated development environment, documentation, drivers, and sample programming code to illustrate how to use the SDK components. Some also include icons and buttons that can easily be incorporated into the design. While it is possible to develop applications without using an SDK, it is much easier using such a powerful resource and so much more can be achieved.

An SDK will include a set of application programming interfaces (APIs) that allows control of the components without the developer needing to know the intricacies of how they work. In some cases, access to the API alone is sufficient to allow significant work to be undertaken, for instance, Eiji Hayashi et al. (2014) only needed access to the APIs.

Assessment Task

Activity No. 1

This activity is the first assignments that together go through the complete development lifecycle for an interactive product.

The goal is to design and evaluate an interactive project. From the redesigning of your eCommerce web page, customer should be able to interact (business transactions like carting, ordering, and paying).

Summary

This chapter examined the requirements activity in greater detail. The data gathering techniques introduced in the previous module are used here in various combinations in the requirements activity. In addition, contextual inquiry, studying documentation, and researching similar products are commonly used techniques. Personas and scenarios help to bring data and requirements to life, and in combination they can be used to explore the user experience and product functionality. Use cases and essential use cases are helpful techniques for documenting the findings from data gathering sessions.

Points to Remember:

- A *requirement* is a statement about an intended product that specifies what it is expected to do or how it will perform.
- Articulating requirements and defining what needs to be built avoids miscommunication and supports technical developers and allows users to contribute more effectively.
- There are different kinds of requirements: functional, data, environmental (context of use), user characteristics, usability goals, and user experience goals.
- Scenarios provide a story-based narrative to explore existing behavior, potential use of new products under development, and futuristic visions of technology use.
- Personas capture characteristics of users that are relevant to the product under development, synthesized from data gathering sessions.
- Scenarios and personas together can be used throughout the product lifecycle.
- Use cases capture details about an existing or imagined interaction between users and the product.

The activities of design, prototyping, and construction. Prototyping and scenarios are used throughout the design process to test ideas for feasibility and user acceptance. We have looked at different forms of prototyping, and the activities have encouraged you to think about and apply prototyping techniques in the design process.

Points to Remember:

- Prototyping may be low fidelity (such as paper-based) or high fidelity (such as software-based).
- High-fidelity prototypes may be vertical or horizontal.
- Low-fidelity prototypes are quick and easy to produce and modify, and they are used in the early stages of design.
- Ready-made software and hardware components support the creation of prototypes.

- There are two aspects to the design activity: conceptual design and concrete design.
- Conceptual design develops an outline of what people can do with a product and what concepts are needed to understand how to interact with it, while concrete design specifies the details of the design such as layout and navigation.
- We have explored three approaches to help you develop an initial conceptual model: interface metaphors, interaction styles, and interface styles.
- An initial conceptual model may be expanded by considering which functions the product will perform (and which the user will perform), how those functions are related, and what information is required to support them.
- Scenarios and prototypes can be used effectively in design to explore ideas.
- Physical computing kits and software development kits facilitate the transition from design to construction.

References

a. Al-Humairi, A., Al-Kindi, O., and Jabeur, N. (2018) Automated Musical Instruments. In *Proceedings of ICMRE 2018 Proceedings of the 2018 4th International Conference on Mechatronics and Robotics Engineering*, pp. 163–169.

b. Banzi, M. (2009) *Getting Started with Arduino*. O'Reilly Media Inc.

c. Gubbels, M., and Froehlich, J. (2014) Physically Computing Physical Computing: Creative Tools for Building with Physical Materials and Computation. *IDC '14 Extended Abstracts*.

d. Greenberg, S., Carpendale, S., Marquardt, N., and Buxton, B. (2012) *Sketching User Experi- ences*. Morgan Kaufmann.

e. Hayashi, E., Maas, M., and Hong, J.I. (2014) Wave to Me: User Identification Using Body Lengths and Natural Gestures, In *Proceedings of CHI 14*, pp. 3453–3462.

f. Hendriks-Jansen, H. (1996) *Catching Ourselves in the Act: Situated Activity, Interactive Emergence, Evolution, and Human Thought*. MIT Press, Cambridge, MA.

g. Making Systems from What's Available, *IEEE Software*, **25**(6), 38–41.

h. Ncube, C., Oberndorf, P., and Kark, A. W. (2008) Opportunistic Software Development:

i. Sharp, H., Preece, J., and Rogers Y. (2019) *Interaction Design: Beyond Human-Computer Interaction - PDF Drive*. (2019). Pdfdrive.com. https://www.pdfdrive.com/interaction-design-beyond-human-computer-interaction-e190044261.html

- END OF MODULE FOR MIDTERM PERIOD –

SUBMISSION OF ASSESSMENT TASKS AND EXAMINATION FOR PRELIM PERIOD WILL BE ANNOUNCED VIA iLearnU LMS or GC

MAKE SURE TO CHECK THE DATES AND DO NOT FORGET TO TAKE IT AS SCHEDULED.