User-centered design

David Lanter
CDM Smith Inc., USA

Rupert Essinger
Esri Inc., USA

User-centered design is a design methodology and philosophy in which the needs, goals, and success of the end user are considered. User-centered design can be applied to any type of system, object or product intended for human use, but the term is used most frequently in connection with computer information system design.

In user-centered design, the ease with which a system can be understood by its intended users, how easily they can complete tasks with it, and how satisfied they are with their use of it, become the key measures of the success of the design. User-centered designs typically require less documentation and training to be successfully used because they have been designed specifically to match the expectations and abilities of the users. Although user-centered design is closely associated with the discipline of user interface design, it can cover almost every aspect of a system, not only its user interface. Elements of a system such as its documentation, how data are entered and structured within the system and made available for use, how it is deployed and presented within an organization, and how the system use fits into the business goals of an organization, can all have user-centered design applied to them.

The term "user-centered" design was made popular by Donald Norman, a cognitive scientist working in the field of system and product usability. He first referred to "user-centered system design" in 1986. His bestselling book, *The Design of Everyday Things* (Norman 1988) (originally published as the *Psychology of Everyday Things*), introduced the concept to a wide audience.

Successful user interfaces meet users' expectations. The application of human factors and ergonomic design principles result in displays and controls that are clearly visible and easy to interact with. User-centered design ensures that displays and controls make sense to the user. A key tenet of user-centered design is that to be intuitive the user interface should match the user's mental model of the task they are performing.

Norman gives the compelling and memorable example of the cooktop found in most kitchens. Cooktops typically have four cooking rings: two at the front nearest the user, and two at the back closer to the kitchen wall. However, the knobs with which the user controls the rings are often arranged in a straight line. This is a bad user interface for this system because there is no immediately obvious relationship (also called "mapping") between the location of the knob and the location of the ring it controls. The spatial arrangement of the knobs does not match that of the rings they control. This basic user interface design flaw makes it hard to use the system and causes the user to frequently turn the wrong ring on.

User-centered design recognizes that the user's ability to understand how to use controls for the cooktop's rings is an important design problem that must be solved before the cooktop is put into production for users to struggle with. When the

The International Encyclopedia of Geography.
Edited by Douglas Richardson, Noel Castree, Michael F. Goodchild, Audrey Kobayashi, Weidong Liu, and Richard A. Marston.

DOI: 10.1002/9781118786352.wbieg0432

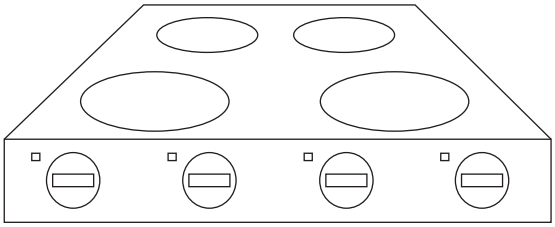

Figure 1 A cooktop with a user interface that is not user-centered. There is not an obvious spatial relationship between the rings and the knobs that control them. Icons are usually added next to each knob to try and make this clearer to the user. (Diagram by Rupert Essinger after Norman.)

four controls are presented in a linear arrangement, there is no obvious relationship of which one controls which ring (Figure 1).

In a kitchen cooktop in which user-centered design has been applied, the controls are arranged as a group in the same pattern as the rings: two in the front and two at the back. This natural mapping of the position of the controls to the position of the rings is immediately understandable by the user because it matches the task as they see it (Figure 2).

When a user interface is not user-centered, documentation is required. In the cooktop with the linear controls, the documentation is usually provided in the form of little icons under each knob designed to show the user which ring each knob controls. These icons are not needed when the knobs are arranged in the same way as the rings: two in the front and two in the back. This illustrates another key tenet of user-centered design: when user interfaces match the user's model successfully, less documentation is required. The goal of user-centered design is to create systems that are successful because they are intuitive and self-explanatory, and that therefore reduces the need for end-user documentation and training.

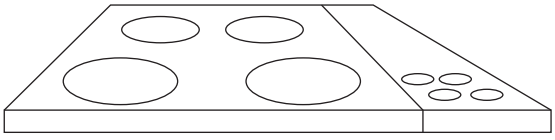

Figure 2 A cooktop in which user-centered design has been applied. It is obvious to the user which control relates to which ring on the cooktop. Icons are, therefore, not required next to the knobs because their spatial arrangement already matches the user's model of the system. (Diagram by Rupert Essinger after Norman.)

It may seem obvious that any system aimed at users should be designed with the user in mind. And especially now that the term "user-centered design" is well known, it is unlikely that a system engineer or designer would admit that their system has not had some amount of user-centered design or at least user-centric thinking applied to it. But modern system development processes vary in the extent to which they involve user-centered design techniques. These techniques include persona-based or role-based design where the needs and workflows of typical users are considered, usability testing of prototypes and development versions of systems with sample users, the involvement of designers with specific usability and human interaction expertise and qualifications, and successive refinement based on feedback from early adopters and beta testers.

User-centered design techniques

Applications and systems may not have one idealized user. User-centered designers realize that they are often not designing for one user, but for a number of different types of users. An application's users will often be segmented by designers into smaller role-based groups, with specific needs based on the information they require and the tasks they want to perform.

Sometimes those needs may pull the user interface design in different directions (Garrett 2003). For example, novices learning new work tasks may find it easier to learn a new application if it presents its workflow as a sequence of simple steps. Experts who have mastered getting the work done, however, may view the sequence of steps enforced by a linear user interface as tedious. These expert users may need a user interface that provides rapid access to a breadth of functions the application provides.

User-centered design recognizes that it may not always be possible to meet both sets of user needs with a single interface solution. One option is to provide different ways for the different user groups to accomplish the same task. Another option is to focus on one group of users and exclude support for the other. This kind of decision has consequences on many of the other choices the designer will need to make as they craft the user interface.

Once an application's users have been identified, the central focus of user-centered design is to provide a solution that solves the users' real problems. The first step involves researching and understanding the activities and tasks that the application's users want to accomplish. This is followed by identifying software features that support those activities. If a feature does not address a real user activity, it does not make it into the user interface design specification. Features that support key user activities get the highest priority in the design specification, and are expected to be both easy to find and easy to accomplish in the user interface.

User-centered design recognizes that users are much better at providing critical feedback in response to a prototype user interface, that is, something they can see, than they are in providing up-front detailed descriptions of what they need or seek. To address this designers often jump-start the process by beginning with nonfunctional mock-ups or storyboards illustrating what the screens could look like, and then change them in response to the users' feedback. The benefits of starting with mock-ups or storyboards include:

- rapid, iterative prototyping to support key user tasks and "use cases";
- prioritization of features and workflows before actual software coding starts;
- users and subject matter experts may feel more comfortable giving feedback on mocked-up prototypes rather than systems that are already in development.

After the mock-ups are implemented, the application will often go through additional usability testing and acceptance testing before it is released, with successive iterations used to refine the user interface and functionality.

Once the application is completed and deployed, user-centered design can still continue. The ability of users to perform their tasks and activities with the system can be reviewed and assessed. Plus, as users gain more experience with the application, they can provide practical feedback and requests for improvements reflecting their actual usage. This is extremely beneficial as it:

- helps focus the priorities and attention of the development team, and is a key input in the planning and design of future iterations of the application;
- supports the refinement and development of any complementary documentation and training both for users and technical support specialists working with them;
- enables managers and user representatives to discuss enhancements to the application in terms of user tasks and roles rather than technical software features;

- allows, once the application is completed and deployed, user-centered design to still continue.

There is a large body of literature about user-centered design, its methodologies, and user interface design. There is also an International Organization for Standardization (ISO) standard dealing with the ergonomics of human-system interaction (ISO 2008).

User-centered design in geographic information systems (GIS)

User-centered design can be applied to geographic information systems (GIS), GIS applications, online mapping systems, and websites designed to provide access to maps, geographic information, and data.

A successful GIS implementation involves not only the use of GIS software, but database design, data acquisition, the development of workflows, best practices and management, data publication, project-based analysis, cartographic design, and application development. User-centered design can be applied to each of these key areas with the goal of ensuring that a system fits with the needs of the people who will use it, rather than the other way round. For example, when an organization plans how data will be acquired, collected, and maintained in its GIS, it will want to ensure that these workflows and policies are easily understandable by the people who will be performing them, and the tools and training provided to different personnel based on their roles, from data collection in the field to database administrator, are successful.

SEE ALSO: Cognition and spatial behavior; Geographic information system

References

Garrett, Jesse James. 2003. *The Elements of User Experience*. Berkeley: New Riders/AIGA.

ISO. 2008. "Ergonomics of Human-System Interaction." ISO 9241-210:2008. Geneva: International Organization for Standardization.

Norman, Donald. 1988. *The Design of Everyday Things*. New York: Basic Books.