Notes - Design - Users and Interaction

- Dr Nick Hayward

A brief intro to users and interaction relative to application and interface design.

Contents

- Intro
- Graphical user interfaces
- Hierarchical breakdown
- Example usage
- Patterns emerging
- Stages of action
- Seven-stage action cycle model
- Resources

Intro   We've considered human vision and how we actually see colours and interfaces. We may now consider some of the processes involved in user interaction itself.

Graphical user interfaces   Graphical user interfaces, GUIs, have tended to present graphical controls such as buttons, drop-down boxes and menus, sliders, and so on. A user is able to interact with the interface of their application either with a sense of directly gesturing on a touchscreen, for example, or indirectly via mouse clicks, keyboard strokes, and other pointing devices.

Inherent in this interface design and user interaction is the assumption that our users know which action to perform to achieve a given application task.

Hierarchical breakdown   User interaction and goal completion within applications usually follows a pretty straightforward, predictable model. This model involves a hierarchical breakdown of their required work into goals, tasks, and actions.

- Goals - a user normally has a high-level goal in mind of what they want to achieve with a given application. This could be writing a letter, taking or modifying a photograph, reading a book, searching for a holiday or flight, chatting with friends, and so on. Goals from this perspective become statements about **what** the user wants to do, instead of **how** they will do it.
- Tasks - to allow our user to fulfill their goals, they will normally have to perform some general steps or follow a structured path of activities that can simply be called **tasks**.
- Actions - our user is able to carry out these tasks by performing **actions** in the interface. Such actions are specific operations in the user interface, and might include clicking a button, selecting a menu item, dragging and dropping, text entry, &c., &c….

Example usage   Let us consider an example application. A user wants to edit some metadata for a photo in their image library application.

This goal might be achieved with some combination of general tasks available within our application.

For example,

- open the required image document
- select a menu item to view the current metadata record
- edit existing text entries in the metadata record
- enter new text for missing data
- spell check user input
- preview the updated image metadata
- tag or categorise the image

and so on.

If we consider one of these general tasks, for example selecting a menu item to view existing metadata, we can start to see some of the **actions** a user might perform in the interface.

- click on a menu item, such as **extras** to select the associated metadata record
- click on the **edit** option to modify the metadata record
- delete some data from the record
- enter some new text
- click the **update** or **save** button to close the metadata record

Patterns emerging The important thing to realise and understand is that a predictable pattern emerges, whereby **goals** can often be achieved by means of various sets of **tasks**, and **tasks** can often be achieved by various sets of **actions**.

We may also note that such interface patterns can be achieved in multiple ways, such as both keyboard shortcuts and mouse inputs, but the pattern from goal to task to action will, more often than not, be the same.

However, it will be necessary to keep such actions flexible and re-usable, allowing us to combine and mix them to achieve multiple disparate tasks.

Stages of action Such stages of action will naturally be easier, and often quicker, for an experienced user. They usually know what tasks are needed to achieve a given goal. They will be able to work out the actions required to accomplish their chosen task. New users will obviously be more hesitant at first, and uncertain of the required actions to accomplish a task. They may even also be uncertain as to which tasks are necessary to achieve their set goal.

Some of these users will consult documentation for the given application, or online tutorials, forums &c., but many users tend to simply begin with an exploratory approach. What happens if I click on this option, or open this particular drawer? **Oh, that happens...** and so on, and so on.

Our user may continue this cycle of exploration through the application, looking for suitable actions, choosing, performing those actions, and then evaluating the end results. In effect, they continue exploring until their current goal has been completed to their satisfaction. Or, until they get stuck and can't move on.

Seven-stage action cycle model So, let us now consider the **Seven-stage action cycle model**.

This process has been developed and elaborated upon by Don Norman in his book **The Design of Everyday Things**, where he describes it quite aptly as the **Seven-stage action cycle model**. It consists of the following steps:

- 1 - Identifying an immediate goal
- 2 - Forming an intention to act
- 3 - Determining a plan of specific actions
- 4 - Carrying out the actions
- 5 - Observing the results by perceiving the state of the system and the world
- 6 - Interpreting the results
- 7 - Evaluating whether the actions had the desired results

These steps are meant to be repeated in an ongoing cycle, with the evaluation of the effects of the actions informing the selection of the next goal. Therefore, this model is considered to describe human-computer interaction as a type of continuous feedback loop between the user and the machine in question.

Resources

- Card, S.K., Moran, T.P. and Newell, A. *The psychology of human-computer interaction.* Lawrence Erlbaum Associates. 1983.
- Krug, S. *Don't make me think, revisited: A common sense approach to web usability.* 3rd Edition. New Riders. 2014.
- Norman, D. *The Design of Everyday Things.* Basic Books. 2013.