Interface for an App—

The design rationale leading to an app that allows someone with Type 1 diabetes to self-manage their condition

Synthesis Lectures on Human-Centered Informatics

Editor

John M. Carroll, *Penn State University*

Human-Centered Informatics (HCI) is the intersection of the cultural, the social, the cognitive, and the aesthetic with computing and information technology. It encompasses a huge range of issues, theories, technologies, designs, tools, environments, and human experiences in knowledge work, recreation and leisure activity, teaching and learning, and the potpourri of everyday life. The series publishes state-of-the-art syntheses, case studies, and tutorials in key areas. It shares the focus of leading international conferences in HCI.

Interface for an App—The design rationale leading to an app that allows someone with Type 1 diabetes to self-manage their condition
Bob Spence

Organizational Implementation: The Design in Use of Information Systems
Morten Hertzum

Data-Driven Personas
Bernard J. Jansen, Joni Salminen, Soon-gyo Jung, and Kathleen Guan

Worth-Focused Design, Book 2: Approaches, Context, and Case Studies
Gilbert Cockton

Worth-Focused Design, Book 1: Balance, Integration, and Generosity
Gilbert Cockton

Statistics for HCI: Making Sense of Quantitative Datas
Alan Dix

Usabiity Testing: A Practitioner's Guide to Evaluating the User Experience
Morten Hertzum

Geographical Design: Spatial Cognition and Geographical Information Science, Second Edition
Stephen C. Hirtle

Human-Computer Interactions in Museums
Eva Hornecker and Luigina Ciolfi

Encounters with HCI Pioneers: A Personal History and Photo Journal
Ben Shneiderman

Social Media and Civic Engagement: History, Theory, and Practice
Scott P. Robertson

The Art of Interaction: What HCI Can Learn from Interactive Art
Ernest Edmonds

Representation, Inclusion, and Innovation: Multidisciplinary Explorations
Clayton Lewis

Research in the Wild
Yvonne Rogers and Paul Marshall

Designing for Gesture and Tangible Interaction
Mary Lou Maher and Lina Lee

From Tool to Partner: The Evolution of Human-Computer Interaction
Jonathan Grudin

Qualitative HCI Research: Going behind the Scenes
Ann Blandford, Dominic Furniss, and Stephann Makri

Learner-Centred Design of Computing Education: Research on Computing for Everyone
Mark Guzdial

The Envisionment and Discovery Collaboratory (EDC): Explorations in Human-Centred Informatics with Tabletop Computing Environments
Ernesto G. Arias, Hal Eden, and Gerhard Fischer

Humanistic HCI
Jeffrey Bardzell and Shaowen Bardzell

The Paradigm Shift to Multimodality in Contemporary Computer Interfaces
Sharon Oviatt and Philip R. Cohen

Multitasking in the Digital Age
Gloria Mark

The Design of Implicit Interactions
Wendy Ju

Interface for an App—The design rationale leading to an app that allows someone with Type 1 diabetes to self-manage their condition
Bob Spence

ISBN: 978-3-031-01105-4 print
ISBN: 978-3-031-02233-3 ebook
ISBN: 978-3-031-00213-7 hardcover

DOI 10.1007/978-3-031-02233-3

A Publication in the Springer series
SYNTHESIS LECTURES ON HUMAN-CENTERED INFORMATICS
Lecture #50
Series Editor: John M. Carroll, Penn State University

Series ISSN 1946-7680 Print 1946-7699 Electronic

Interface for an App—

The design rationale leading to an app that allows someone with Type 1 diabetes to self-manage their condition

Bob Spence
Department of Electrical and Electronic Engineering, Imperial College London

SYNTHESIS LECTURES ON HUMAN-CENTERED INFORMATICS #50

ABSTRACT

This book is an account of how I addressed the need for a smartphone app that would allow someone with Type 1 diabetes to self-manage their condition.

Its presentation highlights the major features of the app's interface design. They include the selection of metaphors appropriate to a user's need to form a mental model of the app; the importance of visible context; the benefits of consistency; and considerations of a user's cognitive and perceptual abilities. The latter is a key feature of the book.

But the book is also about the design process, and especially about the valuable contributions made by the many focus group meetings in which design ideas were first presented to people with Type 1 diabetes. Their critique, and sometimes their rejection, of interface ideas were crucial to the development of the app.

I hope this book will prove useful for teaching and design guidance.

KEYWORDS

App UX design, user interaction, diabetes, app usability, mental model, perception and cognition, requirements

For

LEAH

and

JEREMY

Contents

Terminology

In the following pages we explain some of the terminology that may not be familiar to the reader. They include:

Affordance

Mental models

Metaphors

Dialogues

Exploration

I hope they are useful. I should stress that the rest of the book does not assume that these explanations have been read or understood.

Affordance

FAMILIARITY

Throughout this book the concept of "affordance" is all-pervading, so it is essential to clarify what the term means. Fortunately, this is an easy task, because it is a term that is used quite naturally in conversation. Here are three examples:

- "That door affords (i.e., allows, facilitates) entry to a café."
- "The app must afford (i.e., support) the choice of a favourite meal."
- "The next traffic circle affords access to two motorways."

Let's choose a real example, the door shown in Figure A. A glance at the door, and especially at the vertical plate on its right-hand side, suggests that the door affords opening by pushing on that plate. That glance, together with the assumption about the vertical plate, has identified the door's **perceived affordance**. If we push on that plate and the door opens, we discover its **actual affordance**.

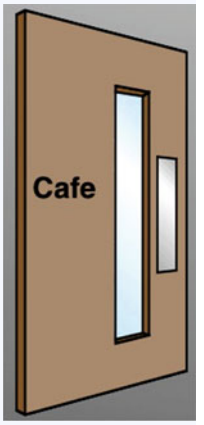


Figure A: This door apparently affords opening by pushing on the vertical plate.

Why do we introduce two definitions of affordance? **Because we want them to be identical**, and that is what a designer—in this case of doors—must try to achieve. Figure B shows another door, whose **perceived affordance** is that one must pull the door open. Only by testing will we know if that is also the **actual affordance**.

A situation that really will cause confusion is shown in Figure C.

Figure B: This door apparently affords opening by pulling on the handle.

Deploying an Affordance

To reach the café behind the door in Figure A, that door's affordance must be **deployed** (i.e., activated, engaged) by some action on the part of the user. For that particular door, the required action is signified by the vertical plate, which is quite reasonably called a **signifier**.

There are many forms that a signifier can take: examples include the vertical plate seen in Figure A and, in the digital domain, familiar icons. In many cases, the design of those icons may not be easy, since their purpose is to ensure that they are correctly interpreted by a user.

Figure C: A very confusing situation!

Signifiers: Do we need them?

But a reasonable question that an interface designer might ask is "should an affordance be accompanied by a signifier?" That question is well illustrated by Figure D, part of a collection of photographs. That presentation may well offer more than one affordance, each deployed by a different user action: horizontal scroll, vertical scroll, mouse click, mouse double-click, continuous mouse-down, and so on. If a visible signifier were to be assigned to all those actions the display would become unacceptably crowded: and that may be the principal reason justifying their absence. Thus, the implicit assumption behind an absence of signifiers may be an interface designer's view that, through training, exploration, friendly advice, or other means, a user will quickly become familiar with the various affordances and not need to interpret signifiers. Not an easy design decision!

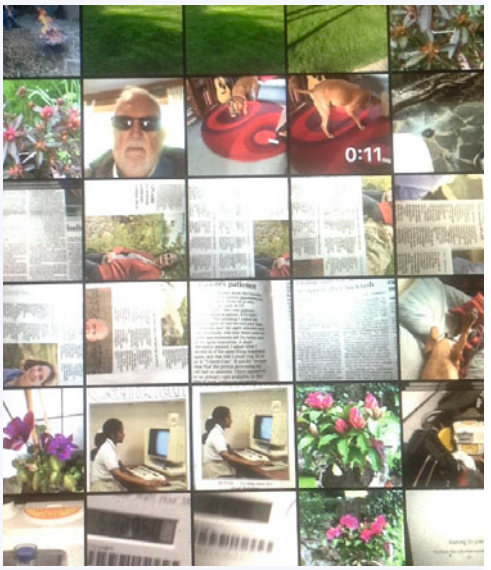


Figure D: Many affordances but an absence of signifiers.

Mental Models and Metaphors

Much is written about mental models, but mostly one is left with the feeling that they are difficult to understand—ephemeral almost. Fortunately, nothing less than a concise definition has been provided by Jacob Neilsen of the Neilsen Norman consulting group:

> **Mental model:** what a user believes they know about a user interface.

Thus, a mental model is based upon **belief**, **not facts**. We also note that, as the user gains experience of an interface, their mental model will be **updated**, and often quite frequently. In other words, a mental model will continually be in a **state of flux**.

Familiarity

Since an app of any complexity will offer many possible interactions, the truly enormous challenge presented to a user-interface designer is now apparent. How can we help a user to form a belief—and preferably a very simple one—about how an app works? An answer is suggested by the term "familiarity". If the property of an interface can be related to something simple and familiar—and often physical—then that property of the interface may be more easily remembered. That "something" can be called a "**metaphor**".

Two Examples

Two illustrative examples of useful metaphors can be taken from the Chapter 4. The first is shown in Figure E: a strip of paper pulled back around two uprights so that all of the paper strip can still be seen. A digital embodiment of that metaphor is in fact used in Chapter 4 to provide a diary for the app user, simply by showing successive days on the "strip of paper". Once explained, not easily forgotten.

The second example—a horizontal stack of transparent vertical plates—is shown in Figure F. On each plate there is a rectangular non-transparent region so sized and positioned that, irrespective of which plate is at the front, no region will ever be totally hidden from view. A digital embodiment simply assigns four important personal attributes (Food, Exercise, etc.) to the four regions, and allows any plate, together with its region, to be brought to "the front" while maintaining the valuable context of the remaining regions.

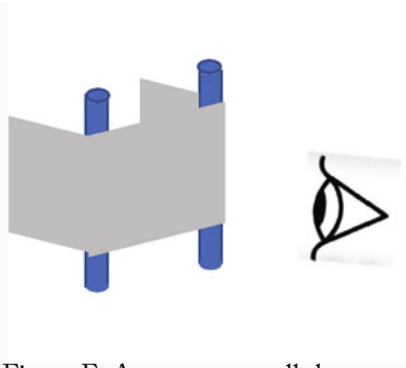


Figure E: A user can see all the paper strip, albeit with some parts distorted.

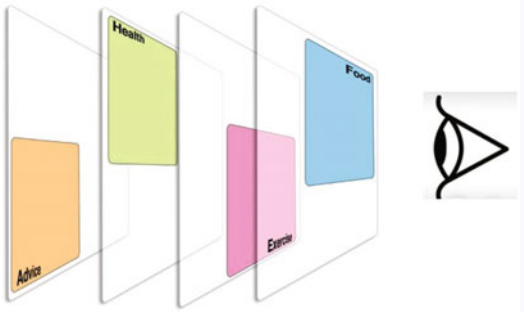


Figure F: A horizontal stack of transparent vertical plates, each containing a non-transparent colored region.

Dialogue

Conversation

When two people engage in a conversation, we say that a **dialogue** is taking place. That dialogue is usually quite complex, containing an unordered, frequently interrupted, and often uncompleted collection of questions, statements, answers, and opinions as well as utterances (e.g., Hmmm, Wow!) of many kinds together with a variety of facial and hand gestures. A documented record of most conversations would make little sense.

Interaction with a Computer

There is no place for such an unstructured dialog between human beings and computers. Nevertheless, even a carefully designed software system can benefit from a check to see if any features could cause a user to experience less than a smooth dialogue. Thus, while there is plenty of guidance regarding interface design, useful checks can instead draw the designer's attention to situations that must be avoided. Below is a list of some situations that should be checked: a more comprehensive list with more extensive discussion can be found in Elmqvist et al. (2011) and within this book (Chapters 5 and 13).

Checks

- Avoid situations in which a user might not notice a change in what has been displayed, a situation that is far more common than expected. Try identifying the difference between the two images at right, for example. A typical way of ensuring that Change Blindness[1] as it is called does not disrupt a dialogue is to ensure that most if not all visual changes are "fluid", ensured by animation of some kind. A duration of about 300 msec is suggested.

[1] See Rensink et al. (1997).

- Ensure that a visual change happens after **every** interaction, employing a visual "shudder" if no change is permitted.
- Ensure that the location of a touch is identical to that of the corresponding visual change. Interactions should be integrated with the visual representation
- Ensure that interaction "never ends". There should be no "dead ends" beyond which interaction cannot proceed.
- To minimize navigational difficulties, provide visual context showing available transitions.
- Ensure that there is an immediate response to every user action.

Those are just some of the useful checks that can be made.

Exploration

Curiosity

Curiosity can take many different forms. From the simple "how much would it cost to borrow £1000?" to the far more challenging "if income tax is raised by 1% what effect, if any, will it have on unemployment?" These and similar questions imply that something might be **changed**, and it is the result of that change that is of interest.

Exploration

Some of the questions regarding change will have specific values in mind. Thus, X will be changed from 3 to 4—what will be the change in Y? In contrast, there are many situations where a user wants to say, "let's vary X and see what happens to Y". They simply want to **explore** the relation between X and Y.

There are two approaches to exploration. In one, the user simply chooses, often arbitrarily, a value of X, waits for Y to be calculated, then chooses a different value for X, and so on. The drawback of this approach to exploration is that the user will have difficulty in remembering the X,Y pairs that have been generated, especially if the underlying calculation takes more than a few seconds.

Dynamic Exploration

The second approach depends upon being able to perform the calculation of Y, given the value of X, in less than about 1 sec, allowing the user to vary X continuously while observing the corresponding values of Y. Such **dynamic exploration**, in which X is smoothly and manually varied, provides the user with a profoundly different experience: within a few seconds they can form a **mental model** of the relation between X and Y.

Predictions

The effectiveness of dynamic exploration requires the calculation of Y within less than a second after X has been specified. If, conventionally, that calculation cannot be carried out so quickly, the enormous benefit that accrues from dynamic exploration nevertheless serves as strong en-

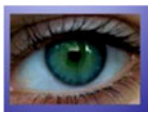
couragement to find a faster means of calculation. Sometimes that will not be possible, and the benefit cannot be achieved. But in many cases, it can—read on!

Example

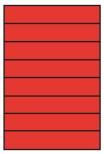
A good example of the benefit of dynamic exploration is provided by the app whose development is the subject of this book. A user with Type-1 diabetes is constantly aware that their blood glucose level (the Y of the calculation mentioned above) must be kept within safe levels, and that it is significantly affected by the carbohydrate value (X) of any meal they consume. Fortunately, for any carbohydrate value, the corresponding blood glucose variation over the next hour or so can be predicted with acceptable confidence. The user will be able, very quickly, to "home in" on a desired approximate range of carbohydrate values which can then be translated flexibly to a meal of interest.

Chapter Summaries

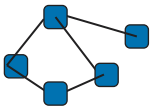
1 ***Introduction*** *This chapter describes the background of the project for which the app design described in this book was created.*



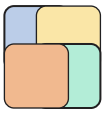
2 ***Requirements*** *In this chapter, I look at how we proceed from the original vision regarding an app to decisions about what activities of the intended user should be supported. A potentially useful and very simple concept in such a development is that of affordance.*

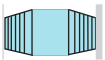


3 ***Structure and layout*** *The requirements identified in Chapter 2 are examined to identify any structure that would benefit design decisions. As a result, a proposal is made regarding the overall general appearance of the smartphone app.*



4 ***Interface metaphors*** *In this chapter we describe two metaphors on which the interface design is based. The main criterion for selecting those metaphors was their ability to help the user form an easily deployed mental model of the interface.*

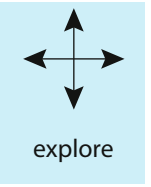




5 ***Dialogue*** *Use of the app will involve the user in several separate interactions, often by touch. While it is essential that each individual interaction should be designed with care, it is also necessary to consider the collection of interactions as a rich dialogue between user and app. Design considerations exist that relate to single interactions, and these are discussed with respect to the Diary and the "stack" of personal regions. Design issues associated with the human visual system and cognition are discussed.*

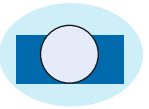
6 ***Data entry*** *The Food affordance comprises seven sub-affordances, one of which ("carb") allows a user to enter the carbohydrate value of a chosen meal and the time of its intended consumption. In this chapter we design the appropriate icon ("signifier") in the Food region that allows this activity to be initiated as well as the tool that is provided to allow those entries to be made.*

7 ***Exploration*** *It is extremely helpful if a user can manually and smoothly vary a carb value and immediately see the effect on predicted blood glucose level over the next hour or so, together with a recommended insulin dose. Such an activity is called "dynamic exploration" and is made possible by our ability, based on machine learning, to make such a prediction.*

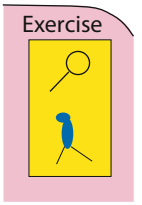


8 ***Favourites*** *We are all creatures of habit, well illustrated by an individual's collection of favourite meals. A user should therefore have easy access to such a collection, and a subordinate affordance "Favourites" provides such access: that access is provided through the convenience of a personalised scrollable menu of favourite meals.*

9 ***Photograph*** *As use of the app proceeds it is anticipated that a user will wish to make additions to their collection of favourite meals. They must therefore be able to photograph such a meal and separately arrange for its appearance in the scrollable collection of Favourites. With consistency with other sub-affordances in mind, we design the tool for such a situation.*



10 ***Exercise*** *Exercise is one of the four personal affordances. Here we design the tool that permits the entry of Exercise parameters such as type (e.g., squash), extent (e.g., 30 min), scheduled time (e.g., 4PM), and aerobic level.*



11 ***Health*** *Health is one of the personal affordances. We design the appearance of its region and the tool that allows appropriate data to be entered. We also use the Health region to illustrate a simple way of determining what temporal data will be presented in the Diary, an approach that generalizes to all personal regions.*

12 ***Advice*** *One essential function of the app is to provide advice to the user. That advice may range from regular recommendations concerning insulin doses to alerts regarding blood glucose levels that require urgent attention and acknowledgement. Conventionally, alerts are often delivered by pop-up windows, but the rationale behind the approach adopted here is that the user's mental model would suffer from such an additional dialogue feature and that the expected location of advice is in the Advice region and its tool.*