

Designing with the Mind in Mind

Simple Guide to Understanding
User Interface Design Rules

Designing with the Mind in Mind

Simple Guide to Understanding
User Interface Design Rules

Jeff Johnson



ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

Morgan Kaufmann Publishers is an imprint of Elsevier



Morgan Kaufmann Publishers is an imprint of Elsevier
30 Corporate Drive, Suite 400, Burlington, MA 01803, USA

This book is printed on acid-free paper

© 2010 Elsevier Inc. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or any information storage and retrieval system, without permission in writing from the publisher. Details on how to seek permission, further information about the Publisher's permissions policies and our arrangements with organizations such as the Copyright Clearance Center and the Copyright Licensing Agency, can be found at our website: www.elsevier.com/permissions.

This book and the individual contributions contained in it are protected under copyright by the Publisher (other than as may be noted herein).

Notices

Knowledge and best practice in this field are constantly changing. As new research and experience broaden our understanding, changes in research methods, professional practices, or medical treatment may become necessary.

Practitioners and researchers must always rely on their own experience and knowledge in evaluating and using any information, methods, compounds, or experiments described herein. In using such information or methods they should be mindful of their own safety and the safety of others, including parties for whom they have a professional responsibility.

To the fullest extent of the law, neither the Publisher nor the authors, contributors, or editors, assume any liability for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions, or ideas contained in the material herein.

Library of Congress Cataloging-in-Publication Data

Johnson, Jeff, Ph. D.

Designing with the mind in mind: simple guide to understanding user interface
design rules / Jeff Johnson.
p. cm.

Includes bibliographical references and index.

ISBN 978-0-12-375030-3 (alk. paper)

1. Graphical user interfaces (Computer systems) I. Title.

QA76.9.U83J634 2010

005.4'37—dc22

2010001844

British Library Cataloguing-in-Publication Data

A catalogue record for this book is available from the British Library.

ISBN: 978-0-12-375030-3

For information on all Morgan Kaufmann publications, visit our
Web site at www.mkp.com or www.elsevierdirect.com

Typeset by MPS Limited, a Macmillan Company, Chennai, India
www.macmillansolutions.com

Printed in China

10 11 12 13 5 4 3 2 1

Working together to grow
libraries in developing countries

www.elsevier.com | www.bookaid.org | www.sabre.org

ELSEVIER

BOOK AID
International

Sabre Foundation

Contents

Acknowledgments.....	vii
Foreword.....	ix
Introduction	xi
CHAPTER 1 We Perceive What We Expect.....	1
CHAPTER 2 Our Vision is Optimized to See Structure.....	11
CHAPTER 3 We Seek and Use Visual Structure	25
CHAPTER 4 Reading is Unnatural	33
CHAPTER 5 Our Color Vision is Limited.....	53
CHAPTER 6 Our Peripheral Vision is Poor	65
CHAPTER 7 Our Attention is Limited; Our Memory is Imperfect.....	79
CHAPTER 8 Limits on Attention, Shape, Thought and Action.....	97
CHAPTER 9 Recognition are Easy; Recall is Hard.....	109
CHAPTER 10 Learning from Experience and Performing Learned Actions are Easy; Problem Solving and Calculation are Hard	119
CHAPTER 11 Many Factors Affect Learning.....	133
CHAPTER 12 We Have Time Requirements	151
Epilogue	173
Appendix.....	175
Bibliography	179
Index	183

Acknowledgments

I could not have written this book without a lot of help and support.

First to mention are the students of the Human-Computer Interaction course I taught as an Erskine Fellow at the University of Canterbury in New Zealand in the winter semester of 2006. It was for them that I developed a lecture providing a brief background in perceptual and cognitive psychology—just enough to enable them to understand and apply user interface design guidelines. That lecture expanded into a professional development course, then into this book.

Second are the reviewers of the first draft: Susan Fowler, Robin Jeffries, Tim McCoy, and Jon Meads. They made many helpful comments and suggestions that allowed me to greatly improve the book.

Third are three cognitive science researchers who provided useful content, directed me to valuable readings, or allowed me to bounce ideas off of them: Prof. Edward Adelson (M.I.T. Dept. of Brain and Cognitive Sciences), Prof. Dan Osherson (Princeton University Dept. of Psychology), and Dr. Dan Bullock (Boston University Dept. of Cognitive and Neural Systems).

The book also was helped immeasurably by the care, oversight, logistical support, and nurturing provided by the staff at Elsevier, especially Mary James, David Bevans, and Andre Cuello.

Valuable additional copyediting was provided by Cate de Heer. Most importantly, I thank my wife and friend Karen Ande for her love and support while I was researching and writing this book, all the more remarkable because it coincided with the period when she was completing a book of her own: *Face To Face: Children of the AIDS Crisis in Africa*, a photography book documenting the plight of children orphaned by AIDS in sub-Saharan Africa (FaceToFaceAfrica.com).

Foreword

The design of interactive computer systems is not only an art, but, at least aspirationally, a science. Well, not a science, actually, but rather a kind of joint computer-cognitive engineering, that is, science-based techniques to create interactive systems satisfying specified requirements.

Like cars, buildings, and clothes, interactive computing artifacts can emotionally delight, exhibit style and fashion, and have social significance. There is much room for art and industrial design in making things that pop, flash, and interact. But the resulting artifacts also have to work correctly and to flow with human activity. A beautiful building whose soaring windows roast its inhabitants in the summer or whose trusses buckle in a storm is a failure. Designers need methods to put latitude, season, fenestration, volume, and circulation together to predict heating loads before building the building. They also need a stockpile of technology component solutions, like thermopane glass, blinds, overhangs, and fans to choose among as part of the standard engineering of a solution. Engineering does not replace art in a design, it makes it possible.

Engineering is hard enough for architecture; it is harder still for interactive artifacts, for the simple reason that it is easier to get a science of buildings than one of people. Providing such a supporting science and engineering has been a founding aspiration of the field of human-computer interaction. How to do it? The most basic method is by “usability testing”—watch users doing tasks, discover their difficulties, and fix these through redesign. Usability testing is useful, necessary, and inefficient. The results don’t cumulate very well into a discipline anything like engineering, and it isn’t very insightful about why things break. It’s the cognitive equivalent of roasting the users to find the effect of the large windows. But usability testing can find many of a system’s flaws. It is a feasible method, because interactive systems are often much easier to change than rebuilding a building.

Better would be to avoid many of the errors in the first place, and one method is through *design rules*. Instead of rediscovering over and over through usability testing that interfaces depending on red and green are bad for color-blind users, just make it a design rule to use color redundantly with other cues. Design rules, however, turn out to have their own problems. In practice, design rules may be ambiguous or require subtle interpretation of context or contradict other guidelines. And that brings us to the current book.

The idea of the present book is to unite design rules with the supporting cognitive and perceptual science that is at their core. This format has several merits: the psychological science is made concrete and easy to absorb by connecting to actual designs, and the design rules are made easier to adjust for context, since they are related to their deeper rationale.

Jeff Johnson is the perfect author to attempt such a book. His whole career has combined work on both the interface design side and the psychological science **ix**

side. I first met him when he was on the user interface team of the Xerox Star series of products—the first commercial graphical user interface. So on the design side, he was essentially in at the beginning of GUIs. On the psychology side, he did degrees at Yale and Stanford. Putting design and psychology together, he worked on commercial interactive systems, taught at the university, and worked as a consultant. His trademark is using concrete design examples to illustrate abstract principles. In fact, he is famous for driving his points home memorably by exhibiting “blooper” examples of bad designs—and so he does in this book.

There is a third method of using science to help engineer a system that goes beyond design rules—*design models*. Jeff’s book shows examples of how to use this method, too. He shows how to model the task context in terms of object and actions and how to understand real-time interaction constraints.

In sum, this is a book that advances the goal of a supporting engineering method for interactive system design. At the same time, it is a primer to understand the *why* of the larger human action principles at work—a sort of cognitive science for designers in a hurry. Above all, this is a book of profound insight into the human mind for practical people who want to get something done.

—Stuart Card

Introduction

USER-INTERFACE DESIGN RULES: WHERE DO THEY COME FROM AND HOW CAN THEY BE USED EFFECTIVELY?

For as long as people have been designing interactive computer systems, some have attempted to promote *good* design by publishing user-interface design guidelines (also called design rules). Early ones included:

- **Cheriton** (1976) proposed user-interface design guidelines for early interactive (time-shared) computer systems.
- **Norman** (1983a, 1983b) presented design rules for software user interfaces based on human cognition, including cognitive errors.
- **Smith and Mosier** (1986) wrote perhaps the most comprehensive set of user-interface design guidelines.
- **Shneiderman** (1987) included “Eight Golden Rules of Interface Design” in the first edition of his book *Designing the User Interface* and in all later editions.
- **Brown** (1988) wrote a book of design guidelines, appropriately titled *Human-Computer Interface Design Guidelines*.
- **Nielsen and Molich** (1990) offered a set of design rules for use in heuristic evaluation of user interfaces.
- **Marcus** (1991) presented guidelines for graphic design in online documents and user interfaces.

In the twenty-first century, additional user interface design guidelines have been offered by Stone *et al.* (2005), Koyani, Bailey, and Nall (2006), Johnson (2007), and Shneiderman and Plaisant (2009). Microsoft, Apple Computer, and Oracle publish guidelines for designing software for their platforms (Apple Computer, 2009; Microsoft Corporation, 2009; Oracle Corporation/Sun Microsystems, 2001).

How valuable are user-interface design guidelines? That depends on who applies them to design problems.

USER EXPERIENCE DESIGN AND EVALUATION REQUIRES UNDERSTANDING AND EXPERIENCE

Following user-interface design guidelines is not as straightforward as following cooking recipes. Design rules often describe goals rather than actions. They are purposefully

very general to make them broadly applicable, but that means that their exact meaning and their applicability to specific design situations is open to interpretation.

Complicating matters further, more than one rule will often seem applicable to a given design situation. In such cases, the applicable design rules often conflict, i.e., they suggest different designs. This requires designers to determine which competing design rule is more applicable to the given situation and should take precedence.

Design problems—even without competing design guidelines—often have multiple conflicting goals. e.g.:

- bright screen *and* long battery life
- lightweight *and* sturdy
- multifunctional *and* easy to learn
- powerful *and* simple
- WYSIWIG (what you see is what you get) *and* usable by blind people

Satisfying all the design goals for a computer-based product or service usually requires tradeoffs—lots and lots of tradeoffs. Finding the right balance point between competing design rules requires further tradeoffs.

Given all of these complications, user-interface design rules and guidelines must be applied thoughtfully, not mindlessly, by people who are skilled in the art of UI design and/or evaluation. User-interface design rules and guidelines are more like *laws* than like *rote recipes*. Just as a set of laws is best applied and interpreted by lawyers and judges who are well versed in the laws, a set of user-interface design guidelines is best applied and interpreted by people who understand the basis for the guidelines and have learned from experience in applying them.

Unfortunately, with a few exceptions (e.g., Norman, 1983a), user-interface design guidelines are provided as simple lists of design edicts with little or no rationale or background.

Furthermore, although many early members of the user-interface design and usability profession had backgrounds in cognitive psychology, most newcomers to the field do not. That makes it difficult for them to apply user-interface design guidelines sensibly.

Providing that rationale and background education is the focus of this book.

COMPARING USER-INTERFACE DESIGN GUIDELINES

Table I.1 places the two best-known user-interface guideline lists side by side to show the types of rules they contain and how they compare to each other (see the Appendix for additional guidelines lists). For example, both lists start with a rule calling for consistency in design. Both lists include a rule about preventing errors. The Nielsen-Molich rule “Help users recognize, diagnose, and recover from errors” corresponds closely to the Shneiderman-Plaisant rule to “Permit easy reversal of actions.” “User control and freedom” corresponds to “Make users feel they are in control.” There is a reason for this similarity, and it isn’t just that later authors were influenced by earlier ones.

Table I.1 The Two Best-Known Lists of User Interface Design Guidelines

Shneiderman (1987); Shneiderman and Plaisant (2009)	Nielsen and Molich (1990)
<ul style="list-style-type: none"> • Strive for consistency • Cater to universal usability • Offer informative feedback • Design task flows to yield closure • Prevent errors • Permit easy reversal of actions • Make users feel <i>they</i> are in control • Minimize short-term memory load 	<ul style="list-style-type: none"> • Consistency and standards • Visibility of system status • Match between system and real world • User control and freedom • Error prevention • Recognition rather than recall • Flexibility and efficiency of use • Aesthetic and minimalist design • Help users recognize, diagnose, and recover from errors • Provide online documentation and help

WHERE DO DESIGN GUIDELINES COME FROM?

For present purposes, the detailed design rules in each set of guidelines, such as those in Table I.1, are less important than what they have in common: their basis and origin. Where did these design rules come from? Were their authors—like clothing fashion designers—simply trying to impose their own personal design tastes on the computer and software industries?

If that were so, the different sets of design rules would be very different from each other as the various authors sought to differentiate themselves from the others. In fact, all of these sets of user-interface design guidelines are quite similar if we ignore differences in wording, emphasis, and the state of computer technology when each set was written. Why?

The answer is that all of the design rules are based on human psychology: how people perceive, learn, reason, remember, and convert intentions into action. Many authors of design guidelines had at least some background in psychology that they applied to computer system design.

For example, Don Norman was a professor, researcher, and prolific author in the field of cognitive psychology long before he began writing about human-computer interaction. Norman's early human-computer design guidelines were based on research—his own and others'—on human cognition. He was especially interested in cognitive errors that people often make and how computer systems can be designed to lessen or eliminate the impact of those errors.

Similarly, other authors of user-interface design guidelines—e.g., Brown, Shneiderman, Nielsen, and Molich—used knowledge of perceptual and cognitive psychology to try to improve the design of usable and useful interactive systems.

Bottom line: user-interface design guidelines are based on human psychology.

By reading this book, you will learn the most important aspects of the psychology underlying user-interface and usability design guidelines.

INTENDED AUDIENCE OF THIS BOOK

This book is intended mainly for software development professionals who have to apply user-interface and interaction design guidelines. This of course includes interaction designers, user-interface designers, and user-experience designers, graphic designers, and hardware product designers. It also includes usability testers and evaluators, who often refer to design heuristics when reviewing software or analyzing observed usage problems.

A second audience for this book is software development managers who want enough of a background in the psychological basis for user-interface design rules to understand and evaluate the work of the people they manage.