

BOOKS FOR REQUIREMENTS ENGINEER

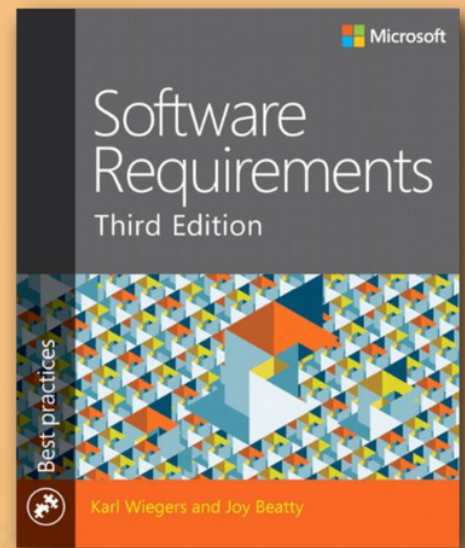
www.lukaspasek.com

I recommend to
read following
books if you want
to learn how to
create good
software
requirements



Karl Wiegers

Software requirements

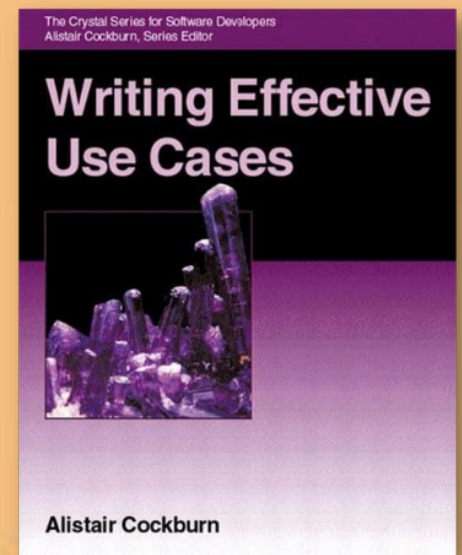


This book teaches the basics of the Requirements Engineer discipline. It is the first book I mention to anyone who thinks about working as a Requirements Engineer. It contains a lot of knowledge coming from a deep and broad understanding of software requirements. I think no one can start applying all the practices mentioned in the book just from reading it one time but you can apply them gradually as you gain more and more experience.

Adds you the following skill: create very good requirements specifications.

Alistair Cockburn

Writing Effective Use Cases



If you think that you do not need software requirements specialist in your project because you have SCRUM or because you follow agile methodology then guess what ... the person who wrote this book is one of the Agile Manifesto authors.

This book is about how to model interaction between humans and computers using so-called 'Use Cases'. This book explains how to think and how to work step by step to model high quality software functionality.

Adds you the following skill: focusing on the end user needs and structuring your software requirements document around those needs.

Soren Lauesen

Task Descriptions as Functional Requirements



This is actually not a whole book but just a single article. However the topic described here helps connect Requirements Engineer with UX Design. This article is so profound because it solves problems which are inherent to Use Cases. Still it is worth reading Alistair Cockburn book and once you do it you will appreciate the beauty of Task Descriptions and you will be prepared to create them. Article by itself may not be enough to be able to create Task Descriptions yourself if you do not have experience with Use Cases.

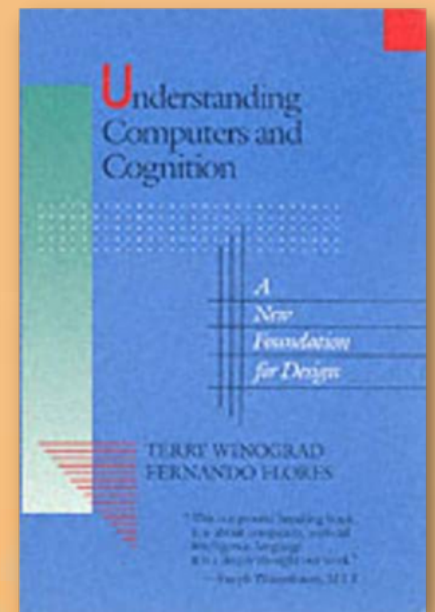
Adds you the following skill: creating a boundary between a problem and its solution.

<https://www.itu.dk/~slauesen/Papers/IEEEtasks.pdf>

www.lukaszpasek.com

Terry Winograd Fernando Flores

Understanding Computers and Cognition: A New Foundation for Design First Printing Edition



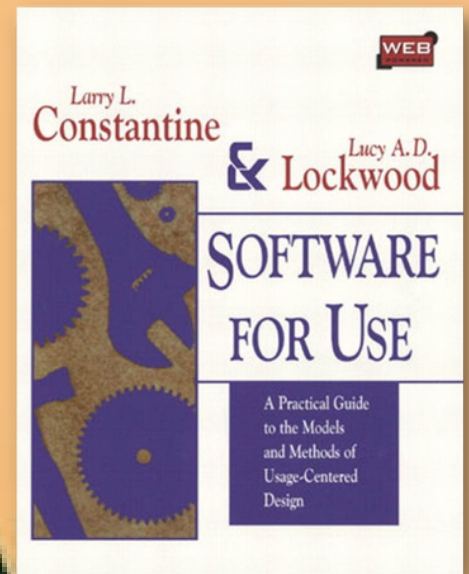
You will find this book very, very interesting today. People who never read it write copious articles all over the internet about how Artificial Intelligence will dominate our world. Well actually it won't and this book explains why it won't.

This book clearly explains what Artificial Intelligence can and can not do. It is quite a complex book which needs time to read. However it proves the superiority of the human mind and body over AI. There are many very interesting science observations which broaden my thinking. After reading this book I understood that anything that can be reached with software depends mostly from the understanding of human interaction with it.

Adds you the following skill: understanding that the decision making shall be on the user side and not the computer or software designer side. This is completely different thinking than most IT professionals have.

Larry L. Constantine Lucy A.D. Lockwood

Software for Use: A Practical Guide to the Models and Methods of Usage-Centered Design



Thanks to this book I understood that usability is not really about look and feel and not about nice fonts, graphics and all other sexy looking stuff. It is not even about colors. It shall all be about the usage and we should treat software as a place where we do work. This book explains how to design a user interface so that it really respects end user time and efforts.

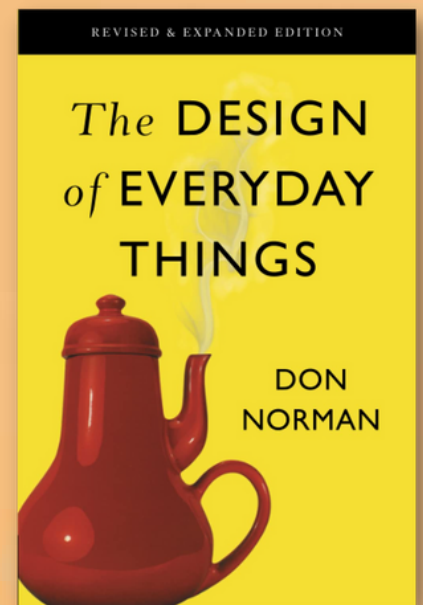
Adds you the following skill: you will be able to create abstract user interface prototypes and you will see beyond user interface prototypes for what they really are.

Donald Norman

The Design Of Everyday Things

This is a great book that explains the basics of usability. I read it as I had known I knew nothing about designing usability into the software. One thing that I understood thanks to this book is that the end user should not be afraid of the user interface. This book explains how to actually get there. First you need to provide undo functionality everywhere in the system. Second you should demonstrate to the user the effects of any functionality before it is used. This in my opinion is not where we are right now with software user interfaces. However we can already be there thanks to applying knowledge from this book.

Adds you the following skill: knowing how to make user interfaces less scary and quick to learn.



Stephen M. McMenamin
John F. Palmer

Essential Systems Analysis

This book has great examples and many diagrams and is really focused on understanding system processing and to think in a clear way about building computer systems. Despite the title which may sound boring, the book is really nice to read. It has very original content which fully explains what processes are and how processes can be connected. One thing you will learn is that one process can not directly connect to another but they need to speak through some kind of data storage. This is a mistake you will see on most Data Flow Diagrams. Understanding this helps you build better software.

Adds you the following skill: you will be able to clearly explain what computer systems are doing and how they can be composed of smaller subsystems.

