

CSE403 Suggested Readings

Rapid Development, Steve McConnell

- 1 - Introduction
- 2 - Rapid-development strategy
- 3 - Classic Mistakes
- 4 - Software Development Fundamentals
- 5 - Risk management
- 6 - Core Issues in Rapid Development
- 7 - Lifecycle planning
- 10 - Customer-Oriented Development
- 12 - Teamwork
- 13 - Team Structure
- 18 - Daily Build and Smoke Test
- 19 - Designing for Change

Anchoring the Software Process, Barry Boehm

<http://citeseer.nj.nec.com/boehm95anchoring.html>

Balancing Discipline and Flexibility with the Spiral Model and MBASE (Model-Based Architecting and Software Engineering), Boehm and Port

<http://www.stsc.hill.af.mil/crosstalk/2001/12/index.html>

Implementing VisiCalc, Bob Frankston

<http://www.frankston.com/public/writing.asp?name=ImplementingVisiCalc>

Painless Functional Specifications, Joel Spolsky

<http://www.joelonsoftware.com/articles/fog0000000036.html>

The Joel Test: 12 Steps to Better Code, Joel Spolsky

<http://www.joelonsoftware.com/printerFriendly/articles/fog0000000043.html>

The Mythical Man-Month, Brooks

Chapter 3, The Surgical Team

The Pragmatic Programmer, Hunt & Thomas

Chapter 7, Before the Project

Structuring Use Cases with Goals, A. Cockburn

<http://alistair.cockburn.us/crystal/articles/alistairsarticles.htm>

Use cases in theory and practice, A. Cockburn

<http://alistair.cockburn.us/crystal/articles/alistairsarticles.htm>

Programming Considered as a Human Activity, EW Dijkstra, Proceedings of the IFIP Congress 65

On the Criteria To Be Used in Decomposing Systems into Modules, DL Parnas, Comm. of the ACM, V15, No 12, Dec 1972

<http://www.acm.org/classics/>

The Hundred-Year Language, Paul Graham

<http://www.paulgraham.com/hundred.html>

Perfection and Simplicity, Taste and Aesthetics, and Designing Distributed Systems, from A Conversation with Ken Arnold, by Bill Venners

<http://www.artima.com/intv/perfect.html>

<http://www.artima.com/intv/taste.html>

<http://www.artima.com/intv/distrib.html>

Software Architecture, David Garlan, CMU, 2001

<http://www-2.cs.cmu.edu/~able/publications/encycSE2001/>

A Practical Method for Documenting Software Architectures, Clements, et al, CMU

<http://www-2.cs.cmu.edu/~able/publications/icse03-dsa/>

If You Didn't Test It, It Doesn't Work, Bob Colwell, IEEE Computer

<http://www.computer.org/computer/homepage/0502/Random/>

Good Enough Quality – Beyond the Buzzword, James Bach

http://www.satisfice.com/articles/good_enough_quality.pdf

The Blind Men and the Elephant (poem), John Godfrey Saxe

<http://www.wordfocus.com/word-act-blindmen.html>

http://www.constitution.org/col/blind_men.htm

The Blind Men and the Elephant – Mastering Project Work, David A. Schmaltz

<http://www.projectcommunity.com/book.html>

Design Patterns Explained: A New Perspective on Object-Oriented Design (2nd ed.), Alan Shalloway and James R. Trott

<http://www.netobjectives.com/dpexplained/index.html>

Mastery – The Keys to Success and Long-Term Fulfillment, George Leonard

<http://www.cs.washington.edu/education/courses/cse403/04wi/mastery.html>

Software Engineering: Ariane 5

http://www2.vuw.ac.nz/staff/stephen_marshall/SE/Failures/SE_Ariane.html