



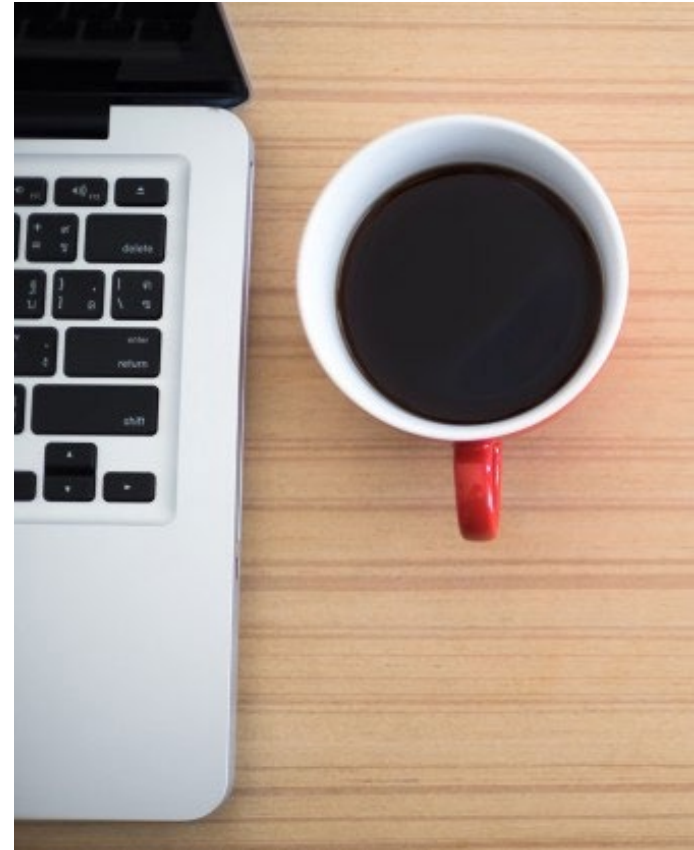
Software Project Management

Course Topics

- Introduction
- Software Process Models
- Requirements Engineering
- Modeling
- Software Construction Techniques
- Testing
- Refactoring
- Project Management
- Ethical Issues

Lecture Objectives

- ✓ Software Project Management
- ✓ Software Management distinctions
- ✓ Software Manager's Roles
- ✓ Project failures
 - [Standish report](#)



Software Project Management



- Concerned with activities involved in ensuring that software is delivered
 - On time/schedule;
 - In accordance with requirements; and
 - Within budget
- Good management cannot guarantee project success
- Bad management usually result in project failure

Client brief



Client budget



Software Management distinctions



- Software product is **intangible**
 - Software cannot be seen or touched. Software project managers cannot see progress by simply looking at the artefact that is being constructed.
 - Manager of a civil engineering project can see the building being developed.
 - Behind the schedule is visible in the actual project, e.g., part of building has not been build.

- Software Manager relies on
 - Reports and reviews to monitor progress.

Software Management distinctions



- Typical engineering process have fixed **standards**.
 - Building bridges, cars etc.
- Software products are uniquely **flexible**.
- Many software projects are ‘**one-off**’ projects.
 - Large software projects are usually different in some ways from previous projects. Even managers who have lots of previous experience may find it difficult to anticipate problems.

Software Manager's Roles

How to lead?

How to organize?

How to collaborate?



How to motivate?

How to create good ideas?

Management Activities



1. Project planning and scheduling.
 - Personnel selection and evaluation.
 - Budgeting
2. Report writing and presentations.
 - To both client and other stakeholders in the company.
3. Risk Management
4. People Management
5. Resource Management
6. Proposal writing.

Standish report (1/4)



- The 2015 CHAOS Report by the Standish Group.
 - published every year since 1994
 - snapshot of the state of the software development industry

- In 2015, report surveyed 50,000 projects around the world, ranging from tiny enhancements to massive systems re-engineering implementations.

Standish report (2/4)

- Following table summarizes the outcomes of projects over the last five years using the new definition of success factors (on time, on budget with a satisfactory result).

MODERN RESOLUTION FOR ALL PROJECTS

	2011	2012	2013	2014	2015
SUCCESSFUL	29%	27%	31%	28%	29%
CHALLENGED	49%	56%	50%	55%	52%
FAILED	22%	17%	19%	17%	19%

The Modern Resolution (OnTime, OnBudget, with a satisfactory result) of all software projects from FY2011–2015 within the new CHAOS database. Please note that for the rest of this report CHAOS Resolution will refer to the Modern Resolution definition not the Traditional Resolution definition.

Standish report (3/4)

- A trend from previous reports that continued in the latest survey is how smaller projects have a much higher likelihood of success than larger ones.

CHAOS RESOLUTION BY PROJECT SIZE			
	SUCCESSFUL	CHALLENGED	FAILED
Grand	2%	7%	17%
Large	6%	17%	24%
Medium	9%	26%	31%
Moderate	21%	32%	17%
Small	62%	16%	11%
TOTAL	100%	100%	100%

The resolution of all software projects by size from FY2011–2015 within the new CHAOS database.

Standish report (4/4)

- Ranking of the factors which work together to make projects more successful.

CHAOS FACTORS OF SUCCESS	
FACTORS OF SUCCESS	POINTS
Executive Sponsorship	15
Emotional Maturity	15
User Involvement	15
Optimization	15
Skilled Resources	10
Standard Architecture	8
Agile Process	7
Modest Execution	6
Project Management Expertise	5
Clear Business Objectives	4

<https://www.infoq.com/articles/standish-chaos-2015>

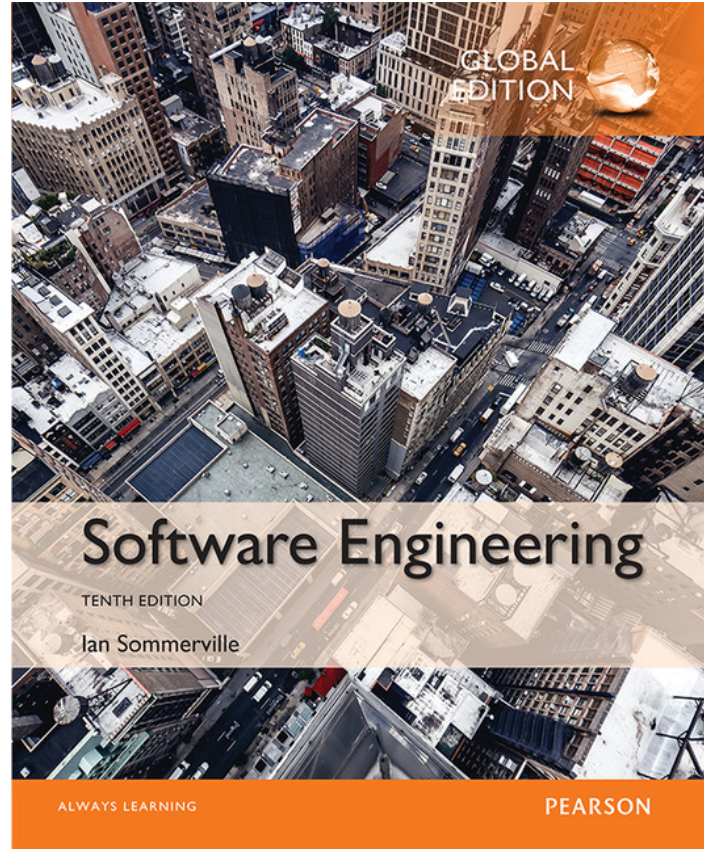
Key Points



- Good software project management is essential if software engineering projects are to be developed on schedule and within budget.
- Software management is distinct from other engineering management.

Read

Chapter 22



References



- Ian Sommerville, “Software Engineering”, 10th Edition, Addison-Wesley, 2015.
- Timothy C. Lethbridge and Robert Laganière, “Object-Oriented Software Engineering: Practical Software Development using UML and Java”, 2nd Edition, McGraw Hill, 2001.
- R. S. Pressman, Software Engineering: A Practitioner’s Approach, 10th Edition, McGraw-Hill, 2005.
- <https://www.standishgroup.com/>