

Software Project Management (Second Edition)

Bob Hughes and Mike Cotterell,
School of Information Management, University of Brighton

The McGraw-Hill Companies

London • Burr Ridge, IL • New York • St Louis • San Francisco • Auckland • Bogotá Caracas
Lisbon • Madrid • Mexico • Milan • Montreal • New Delhi • Panama • Paris • San Juan • São Paulo
Singapore • Tokyo • Toronto

Contents

1	Introduction to software project management	1
1.1	Introduction	1
1.2	What is a project?	2
1.3	Software projects versus other types of project	3
1.4	Activities covered by software project management	3
1.5	Some ways of categorizing software projects	6
1.6	The project as a system	7
1.7	What is management?	8
1.8	Problems with software projects	9
1.9	Management control	11
1.10	Stakeholders	13
1.11	Requirement specification	14
1.12	Information and control in organizations	15
1.13	Conclusion	17
1.14	Further exercises	18
2	Step Wise: an overview of project planning	19
2.1	Introduction to Step Wise project planning	19
2.2	Step 0: Select project	20
2.3	Step 1: Identify project scope and objectives	22
2.4	Step 2: Identify project infrastructure	24
2.5	Step 3: Analyse project characteristics	27
2.6	Step 4: Identify project products and activities	28
2.7	Step 5: Estimate effort for each activity	32
2.8	Step 6: Identify activity risks	32
2.9	Step 7: Allocate resources	33
2.10	Step 8: Review/publicize plan	34
2.11	Steps 9 and 10: Execute plan and Lower levels of planning	35
2.12	Conclusion	35
2.13	Further Exercises	36
3	Project evaluation	37
3.1	Introduction	37
3.2	Strategic assessment	38
3.3	Technical assessment	40

3.4	Cost–benefit analysis	40
3.5	Cash flow forecasting	42
3.6	Cost–benefit evaluation techniques	43
3.7	Risk evaluation	50
3.8	Conclusion	55
3.9	Further exercises	55
4	Selection of an appropriate project approach	57
4.1	Introduction	57
4.2	Choosing technologies	59
4.3	Technical plan contents list	63
4.4	Choice of process models	63
4.5	Structured methods	64
4.6	Rapid application development	64
4.7	The waterfall model	65
4.8	The V-process model	66
4.9	The spiral model	67
4.10	Software prototyping	67
4.11	Other ways of categorizing prototypes	70
4.12	Tools	71
4.13	A prototyping example	72
4.14	Incremental delivery	73
4.15	An incremental example	76
4.16	Selecting the most appropriate process model	76
4.17	Conclusion	77
4.18	Further exercises	77
5	Software effort estimation	79
5.1	Introduction	79
5.2	Where are estimates done?	81
5.3	Problems with over- and under-estimates	82
5.4	The basis for software estimating	84
5.5	Software effort estimation techniques	85
5.6	Expert judgement	87
5.7	Estimating by analogy	88
5.8	Albrecht function point analysis	89
5.9	Function points Mark II	92
5.10	Object points	94
5.11	A procedural code-oriented approach	96
5.12	COCOMO: a parametric model	97
5.13	Conclusion	103
5.14	Additional exercises	104

8.8	Publishing the resource schedule	162
8.9	Cost schedules	164
8.10	The scheduling sequence	165
8.11	Conclusion	167
8.12	Further exercises	167
9	Monitoring and control	169
9.1	Introduction	169
9.2	Creating the framework	169
9.3	Collecting the data	173
9.4	Visualizing progress	175
9.5	Cost monitoring	179
9.6	Earned Value	180
9.7	Prioritizing monitoring	185
9.8	Getting the project back to target	186
9.9	Change control	188
9.10	Conclusions	190
9.11	Further exercises	190
10	Managing contracts	191
10.1	Introduction	191
10.2	Types of contract	192
10.3	Stages in contract placement	198
10.4	Typical terms of a contract	203
10.5	Contract management	206
10.6	Acceptance	208
10.7	Summary	208
10.8	Further exercises	208
11	Managing people and organizing teams	211
11.1	Introduction	211
11.2	Understanding behaviour	213
11.3	Organizational behaviour: a background	213
11.4	Selecting the right person for the job	215
11.5	Instruction in the best methods	217
11.6	Motivation	217
11.7	Working in groups	221
11.8	Becoming a team	221
11.9	Decision making	224
11.10	Leadership	226
11.11	Organizational structures	229
11.12	Conclusion	232
11.13	Further exercises	232

8.8	Publishing the resource schedule	162
8.9	Cost schedules	164
8.10	The scheduling sequence	165
8.11	Conclusion	167
8.12	Further exercises	167
9	Monitoring and control	169
9.1	Introduction	169
9.2	Creating the framework	169
9.3	Collecting the data	173
9.4	Visualizing progress	175
9.5	Cost monitoring	179
9.6	Earned Value	180
9.7	Prioritizing monitoring	185
9.8	Getting the project back to target	186
9.9	Change control	188
9.10	Conclusions	190
9.11	Further exercises	190
10	Managing contracts	191
10.1	Introduction	191
10.2	Types of contract	192
10.3	Stages in contract placement	198
10.4	Typical terms of a contract	203
10.5	Contract management	206
10.6	Acceptance	208
10.7	Summary	208
10.8	Further exercises	208
11	Managing people and organizing teams	211
11.1	Introduction	211
11.2	Understanding behaviour	213
11.3	Organizational behaviour: a background	213
11.4	Selecting the right person for the job	215
11.5	Instruction in the best methods	217
11.6	Motivation	217
11.7	Working in groups	221
11.8	Becoming a team	221
11.9	Decision making	224
11.10	Leadership	226
11.11	Organizational structures	229
11.12	Conclusion	232
11.13	Further exercises	232

C.3	An overview of the EM acquisition process	292
C.4	Acquisition goal definition	292
C.5	Acquisition planning	293
C.6	Procurement	295
C.7	Adaptation planning	295
C.8	Method bridging	300
C.9	Conclusions	301
D	ISO 12207 – an overview	303
D.1	Introduction	303
D.2	The ISO 12207 approach to software life cycle data	303
D.3	The ISO 12207 approach to software life cycle processes	304
D.4	The acquisition process	305
D.5	The supply process	308
D.6	The development process	309
E	Project Management Bodies of Knowledge	315
E.1	Introduction	315
E.2	Project Management Institute	316
E.3	Australian Institute of Project Management	319
E.4	Association for Project Management	320
E.5	UK National Vocational Qualifications	322
E.6	Information Systems Examination Board	324
F	Answer pointers	327
	Further reading	367

Preface to the second edition

Since the first edition of *Software Project Management*, the perception of the importance of project management and consequently its development as a discipline has continued to grow. In the UK, for example, we have seen the publication of the British Standards Institution guidelines on project management (BS 6079) and a revised and improved version of the PRINCE standard. The British Computer Society professional examinations now include a paper on project management. The Association for Project Management has continued to develop its Body of Knowledge and its system of qualifications, while the Project Management Institute in the United States, through its seminal Body of Knowledge document, is making its influence felt world-wide. In a European context, *Euromethod* has been published. This initiative attempts, admittedly with mixed success, to address top-level project management issues. We have also been made aware of the impressive set of national vocational competence standards on project management that have been developed in Australia.

Our contacts with industry suggest that part of this growing interest in project management is because organizations have become 'leaner' and 'delayed', so that the burden of keeping businesses running has been put on the shoulders of a smaller and more hard-pressed work force. Staff who might regard themselves as primarily technical people often find that 'empowerment' means that they now have to plan and manage work where previously this would have been done for them. The removal of layers of management also means that organizational changes often require a project approach where previously they could have been implemented as part of normal day-to-day organizational management.

We have found some who have claimed that the managers of IT projects do not need to have any specific expertise in IT matters: that essentially there is no need for software project management. As the title of this book indicates, we are not of this view. As Darryl Ince of the Open University has noted, software disasters since 1995 have not abated and if anything have increased, especially where client-server software has been the subject of development. It seems clear that project managers need to be aware of the issues and problems of IT development and IT developers need to have project management skills.

The target audience for this book remains students of disciplines such as information technology, information systems and computer science where project management is part of their course; and also practitioners, typically IT developers who have just or are about to assume project management responsibilities.

Preface to the first edition

The effective management of projects in IT environments has increasingly been seen to be important in recent years. In the UK, some aspects of this have been:

- the development of the government-sponsored PRINCE standard
- the setting up of the PROMS-G project management special interest group of the British Computer Society
- the provision of a Certificate of Proficiency in project management by the Information Systems Examinations Board.

Our contacts with industry and commerce have underlined for us the significance of what might be regarded as very basic project management measures. It is our belief that these fundamental practices need to be stressed so that they become first nature for IT practitioners, as a very important part of the foundations of their professional education. While it is hoped that there may be something of interest in this book for the experienced project manager, it is targeted more directly at students about to enter the world of IT development, either through an industrial placement or a first job, and those already in software development who are just starting to take on project management responsibilities and who seek guidance.

Bearing in mind this audience, we have made extensive reference to two imaginary scenarios which explore the concerns of two new project leaders, Amanda and Brigitte, who are undertaking their first project management roles.

We touch upon many techniques that may be of assistance when planning and controlling projects. Some of these may be more appropriate than others in particular circumstances. To give coherence we have provided Step Wise, a general framework compatible with PRINCE, for project planning into which the various techniques can be fitted.

We would especially like to thank Ken I'Anson of the CCTA for his helpful suggestions about the Appendix on PRINCE, and also Dave Hatter and Chris Clare for their guidance. The early advice of David Howe and Martin Campbell Kelly on the basic content and format of this book is also gratefully acknowledged. Thanks, too, to Barbara Kitchenham of the NCC, Manchester for her permission to us to use a project data set shown in the Chapter 5. We know that we have picked up many ideas from our colleagues and we acknowledge this particularly in the case of John Williams, Garth Glynn and Heinz Seefried. The work of putting