

## PREFACE

Why Computing (as an academic discipline) does projects

Why Computing departments do projects

Ways to use this book

    Knowing what's out there

    Knowing what is possible, what you can change

    Knowing How to Change

    If you're not in the Computing discipline

The parts of this book

References

Acknowledgements

## PART ONE

Introduction

What is a composite case study?

What are mechanisms?

What are specific case studies?

Project models

Project choice

### One: COMPOSITE CASE STUDIES

- 1.1 Final-year individual project
- 1.2 Second-year group project
- 1.3 Taught M.Sc. project
- 1.4 Project with handover (a.k.a. 'Software Hut')
- 1.5 Research-type project
- 1.6 Design-and-build project
- 1.7 Project with industrial involvement
- 1.8 Project with a client
- 1.9 Process-based project
- 1.10 Integrative or "capstone" project
- 1.11 The professional bodies' view

### Two: MECHANISMS

- 2.1 Allocation of topics to students (or teams of students)
- 2.2 Allocation of students (or teams) to supervisors
- 2.3 Allocation of students to teams for group projects
- 2.4 Supervisor's roles
- 2.5 Meeting composition: attendance and focus
- 2.6 Time for meetings
- 2.7 Roles in groups
- 2.8 Motivation
- 2.9 (i) Nature of assessment
- 2.9 (ii) Group assessment
- 2.9 (iii) Basis of assessment: deliverables
- 2.9 (iv) Assessment: who marks
- 2.10 Marking schemes
- 2.11 Overseeing, moderation and QA
- 2.12 Staff deployment

### Three: SPECIFIC CASE STUDIES

Introduction

- 3.1 Large-scale group project (University of York, UK)
- 3.2 Project managed by negotiation (University of Teesside, UK)
- 3.3 Creating a real company (University of Sheffield, UK)

- 3.4 3rd year students supervising 1st year groups (University of Leeds, UK)
- 3.5 Emphasis on Personal Transferable Skills (University of Exeter, UK)
- 3.6 International group project (Uppsala University, Sweden and Grand Valley State University, US)
- 3.7 Computing history projects (Metropolitan State College of Denver, US)