Aston University
BIRMINGHAM UK

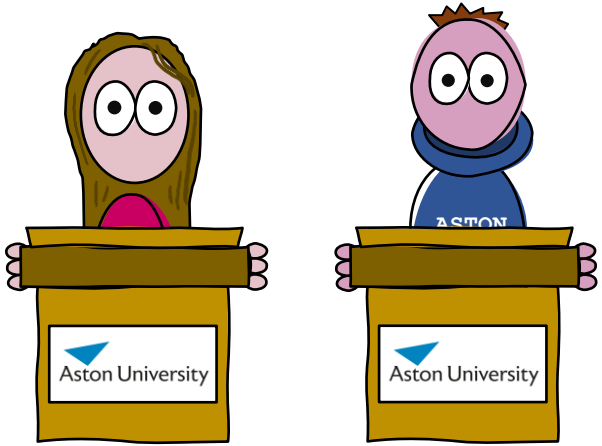
Software Project Management

Unit 1: Introduction (part 1)

Thais Webber
Richard Lee

Module staff

- Module tutors:
  - Thais Webber
  - Richard Lee
- Details can be found on Blackboard:
  - Links to WASS calendars
  - e-mail addresses
- Module information
  - Module Specification CS3SPM
  - Module Overview (our planned lectures/tutorials)



Module Overview (Blackboard)

(2025-26 CS3SPM) Software Project Management

Module Overview

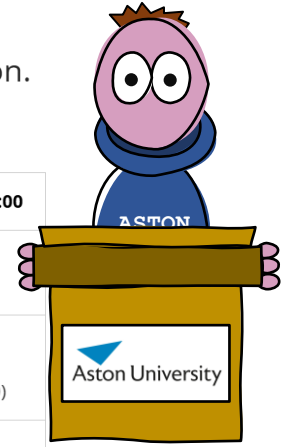
Visible to students

Edit content

Welcome to CS3SPM module!

Below you find our weekly content available and scheduled tutorial sessions information.

Week#	W/C (online) 9:00-11:00	Content	Tutorial (in-person): 10:00-11:00
1	22-Sep-2025	Introduction to Software Project Management	-
2	29-Sept-2025	Measurement, estimation & data analysis (part 1)	Measurement tutorial 03-Oct-2025 - Great Hall (450)
3	06-Oct-2025	Measurement, estimation & data analysis (part 2)	Estimation tutorial 10-Oct-2025 - Great Hall (450)



Motivation

Why Software Project Management module?

- Who has started the Final Year Project (FYP) CS3IP?

Motivation

Aston University
BIRMINGHAM UK

Why Software Project Management module?

- Who has started the Final Year Project (FYP) CS3IP?

- Do you know that one of the main reasons of failure of the FYP is …
  - the lack of good strategies to manage the time to develop the project (e.g., following a plan and updating it when required)?
  - the lack of communication with stakeholders (e.g., supervisor, tutor)?

How the customer explained it

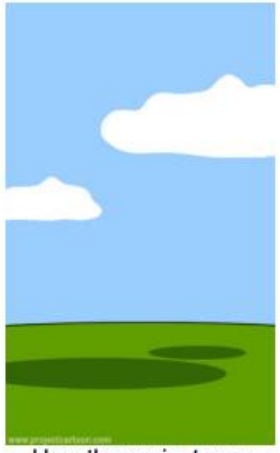
How the project leader understood it

How the analyst designed it

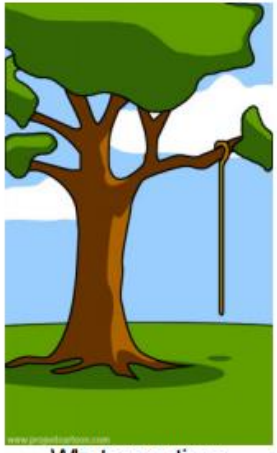
How the programmer wrote it

What the beta testers received

How the business consultant described it

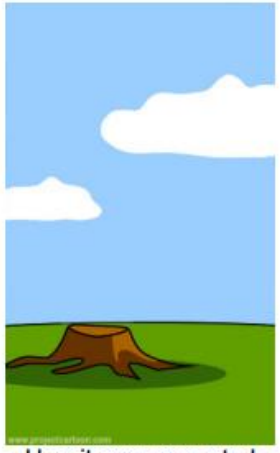


How the project was documented



What operations installed

How the customer was billed



How it was supported

What marketing advertised

What the customer really needed

What about your Final Year Project?

- Prepare yourself in advance. Ask yourself:
  - What should you do to succeed?
  - What are the risks to avoid?
  - What are the best strategies to follow?
  - What to do if you are confronted with a problem?

- This module will help you to answer those questions

Unit 1: Introduction to Software Project Management

Aston University
BIRMINGHAM UK

When you have completed this unit, you will be able to:

- Appreciate the need for a project management module dedicated to software development
- Understand the aims, scope and structure of the module
- Define key concepts and terminology associated with software project management

Module information

Aston University
BIRMINGHAM UK

- Exploring the underlying principles of software project management, and their use in quantitative approaches to managing software projects (Throughout the module)
- How to define software metrics and how to measure, estimate and analyse data associated with software projects (Weeks 2-3)
- How to plan and schedule resources in software projects (Weeks 4-5)
- How to manage, monitor and control risk, quality and people in software projects (Weeks 6-10)

Need for principled, quantitative approaches

Aston University
BIRMINGHAM UK

Example 1*:

As the software manager responsible for building a new system, you are required to tell the sales team when the system will be ready. You have 25 use cases as requirements, and 5 software engineers who tell you that they can have the system ready to ship in four months.

Do you accept this estimate or not?

**Adapted from Laird & Brennan, Software Measurement and Estimation, Wiley, 2006.*

Need for principled, quantitative approaches

Aston University
BIRMINGHAM UK

Example 2*:

You are responsible for making a go/no-go decision on releasing a new software application. The testing team tells you that there are approximately eight defects per thousand lines of code left in the system.

Should you say ‘yes’ or ‘no’?

**Adapted from Laird & Brennan, Software Measurement and Estimation, Wiley, 2006.*

This module will provide the knowledge and tools required to take these decisions with confidence, based on rigorous analysis of available data

Why a module dedicated to *software* project management?

Aston University
BIRMINGHAM UK

- Software is a relatively new field
  - decades old, compared to thousands of years for building bridges/houses
- Subject to rapid change in tools and techniques
- Difficult to learn from experience when all projects are unique
- Lack of physical artefacts
- Large number of unsuccessful / late / over-budget projects

What makes a successful project?

- User involvement
- Executive management support
- Clear statement of requirements
- Proper planning
- Realistic expectations

For reflection…
What are some examples of unrealistic expectations in a project?

Topics CS3SPM

Aston University
BIRMINGHAM UK

- Introduction (Week 1)
- Measurement, Estimation & Data Analysis (Weeks 2-3)
- Software Project Planning (Weeks 4-5)
- Agile vs Traditional Methodologies (Week 6)
- Risk Management (Week 8)
- Monitoring, Control and Quality Management (Week9)
- Revision (Week 10)

**On CS3SPM Blackboard page:*

> Module Information > Module Overview

Mode of learning

- Lecture
  - Mondays, 9-11am
  - Online
  - Recorded and slides provided

- Tutorial
  - (some) Fridays, 10-11am (in-person)
  - On-campus, room can vary (check your timetable)
  - Run by a lecturer and teaching assistants
  - Problem solving based on exam-type questions
  - Questions ideally should be answered in advance
  - Solutions are provided after each tutorial

Assessment

- 100% closed-book exam (in January):
  - Questions on theoretical aspects of Software Project Management covered in lectures and tutorials
  - Problems similar to those addressed in tutorials and revision session
  - Exam contains sections
    - A mandatory section (general questions on all topics covered)
    - A section comprising a choice of questions
  - A guide for self-study for preparation for the exam (and quizzes) will be made available on Blackboard and reviewed in Week 10 (last lecture).

https://www.aston.ac.uk/academic-services/for-students/examinations/examination-calendars

Feedback

Aston University
BIRMINGHAM UK

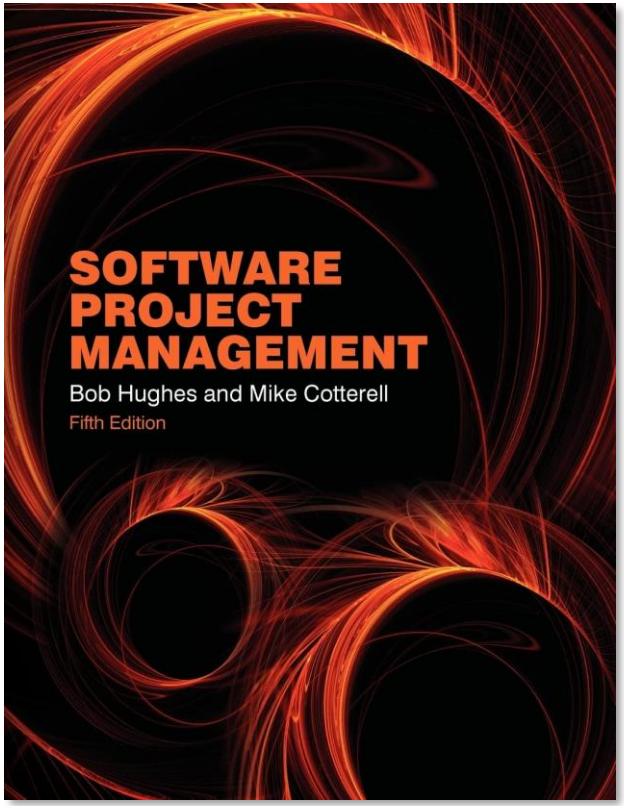
- Formal feedback:
  - Example solutions for tutorial problems
- Informal feedback:
  - Discussions during tutorials
  - Questions answered during lectures and labs
  - Office hours

Textbook

- Bob Hughes & Mike Cotterell, *Software Project Management*, 5th edition, McGraw Hill, 2009

- Essential for this module, available from the library as a physical book and eBook

- *Other resources*
  - *Blackboard Module information*
  - *Please check >Talis Reading list*



Additional (recommended) reading

Aston University
BIRMINGHAM UK

1. The art of scrum : how Scrum masters bind dev teams and unleash agility. Book author: Dave McKenna, 2016.
2. Agile project management with Kanban. Book author: Eric Brechner, 2015.
3. The Mythical Man-Month. Frederick Brooks (author), 1995; latest Essays on Software Engineering, 2021 ed.

Additional (further) reading

1. Linda M. Laird & M. Carol Brennan, *Software Measurement and Estimation: A Practical Approach*, Wiley-Blackwell, 2006.

2. James Cadle & Donald Yeates, *Project Management for Information Systems*, 5th edition, Prentice Hall, 2007.

3. Norman E. Fenton & Sharli L. Pfleeger, *Software Metrics: A Rigorous and Practical Approach*, 2nd edition, PWS Publishing, 1998.

4. Shari L. Pfleeger & Joanne M. Atlee, *Software Engineering: Theory and Practice*, 4th edition, Prentice Hall, 2009.

Aston University
BIRMINGHAM UK

Software Project Management

Unit 1: Introduction (part 2)

Thais Webber
Richard Lee

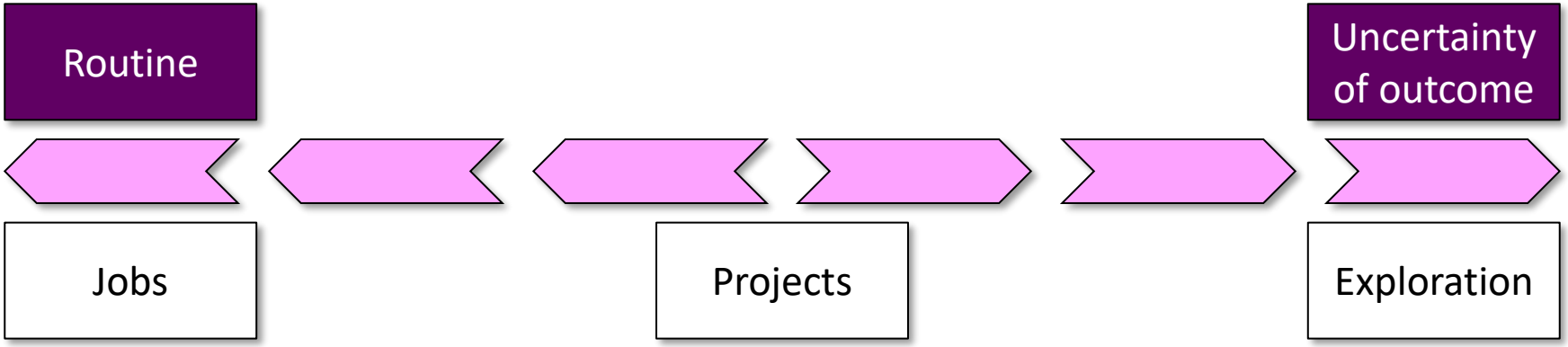
What is a project?

Aston University
BIRMINGHAM UK

- Some dictionary definitions:
  - A specific plan or design
  - A planned undertaking
  - A large undertaking, e.g., a public works scheme

- Key points above are **planning** and **size** of task

Jobs versus projects



- **Jobs**: repetition of well-defined, well-understood tasks with little or no uncertainty
- **Exploration**: e.g., finding a cure for a disease; the outcome, duration and budget are highly uncertain
- **Projects**: lie in the middle of both

Characteristics of projects

- Non-routine
- Planned
- Aiming at a specific target
- Carried out for a customer
- Carried out by a temporary work group
- Involving several specialisms
- Made up of several different phases
- Constrained by time and resources
- Large and/or complex

Some ways of categorising projects

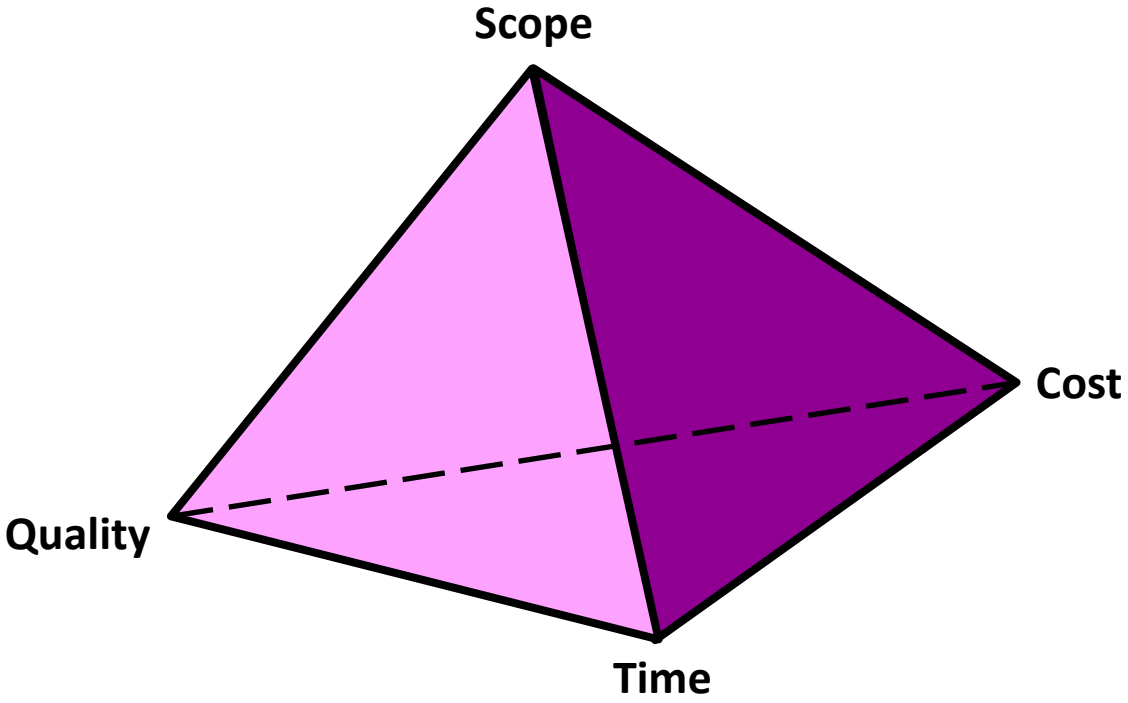
- Voluntary systems (e.g., computer games) *versus*
  Compulsory systems (e.g. the order processing system in an organization)

- Information systems (used by human operators directly) *versus*
  Embedded systems (controlling a device or machine)

- Objective-based (client provides problem that can be solved in multiple ways)
  versus Product-based (client provides product specification)

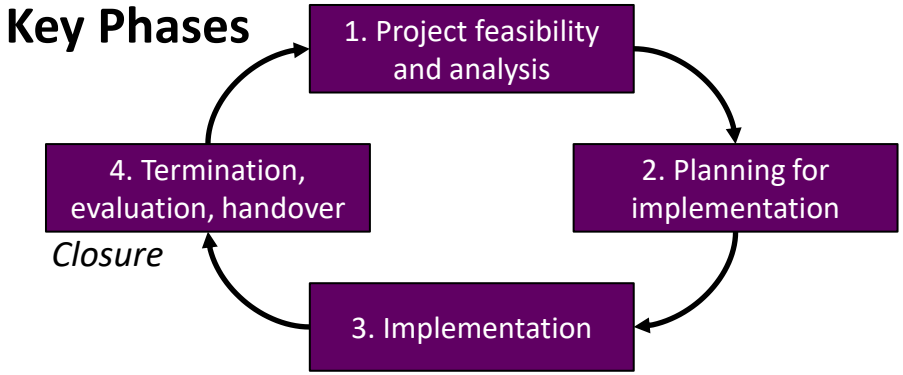
Distinguishing different types of project is important as different types of tasks need different project approaches

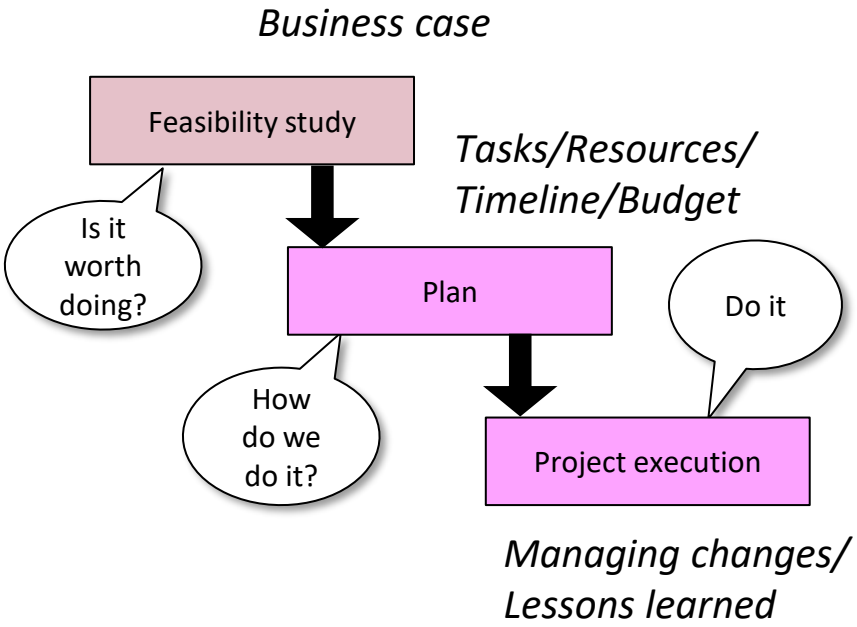
Project factors in management



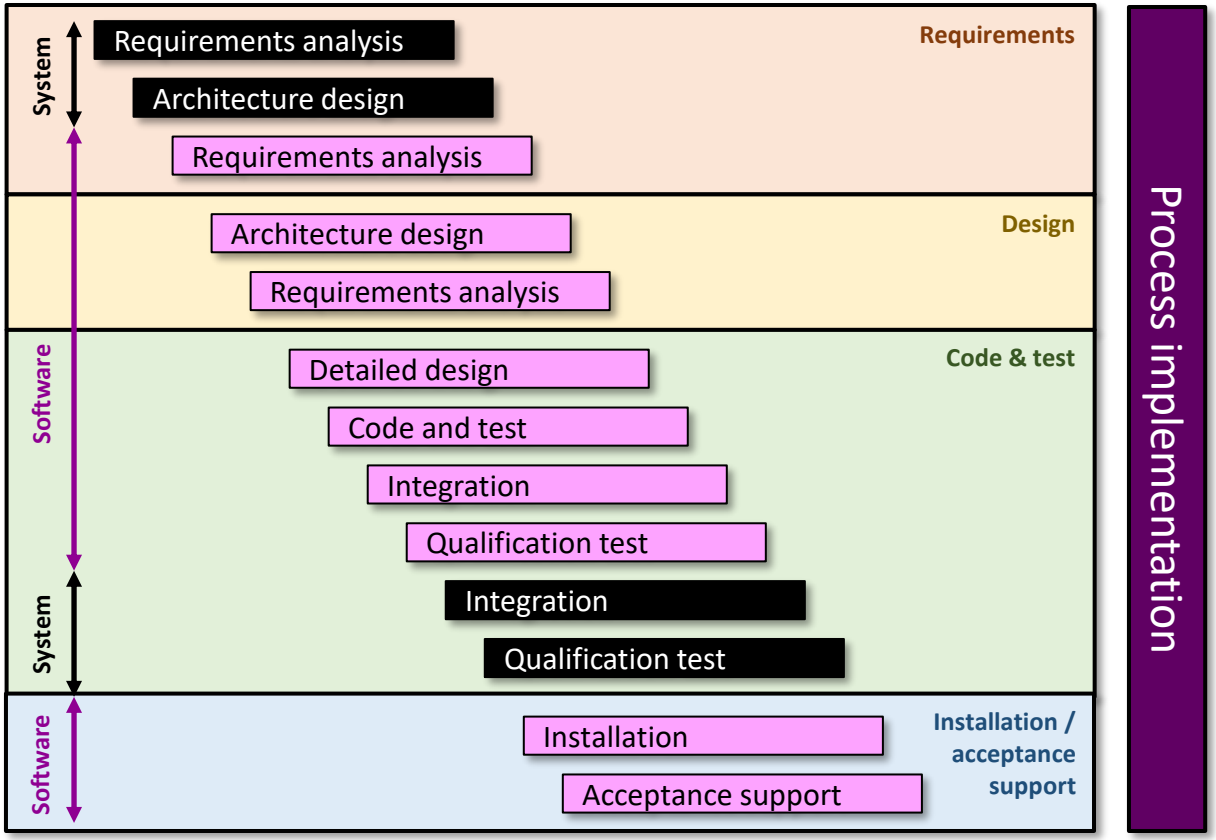
Project management cycle

Source: book SPM pg.5





Software lifecycle (ISO 12207)



*A new ISO/IEC/IEEE draft update (P12207) is in development

ISO 12207: requirements

Aston University
BIRMINGHAM UK

- **Requirements elicitation**: what does the client need?

- **Requirements analysis**: converting ‘customer-facing’ requirements into equivalents that developers can understand

- **Types of requirements**
  - **Functional** requirements: What the software should do—its main features or functions.
  - **Non-functional** requirements: How well the software should work—for example, its speed (performance), reliability, security, and quality.
  - **Resource constraints**: Limitations or restrictions, like cost (budget), deadlines, and technology choices.

ISO 12207: design and code & test

Aston University
BIRMINGHAM UK

- **Architecture design**
  - Based on system requirements
  - Defines components of system: hardware, software, organizational
  - Software requirements will come out of this
- **Code and test**
  - Of individual components
- **Integration**
  - Putting the components together

ISO 12207 continued

Aston University
BIRMINGHAM UK

- Qualification testing
  - Testing the **system** (not just the **software**)
- Installation
  - The process of making the system operational
  - Includes setting up standing data, setting system parameters, installing on operational hardware platforms, user training, etc.
- Acceptance and Support
  - Including maintenance and enhancement

What is a stakeholder?

Aston University
BIRMINGHAM UK

A stakeholder is any person or organisation impacted by, or able to influence, a software project's requirements or outcome.

- **Why they matter**
  - Shape requirements
  - Spot problems early
  - Provide feedback
  - Ensure product is useful and adopted
  - Missing stakeholders: unmet needs & higher risk
- **Different stakeholders may have different/conflicting objectives**
  - Need to define common project objectives
  - Need to agree on an acceptable compromise

Stakeholders

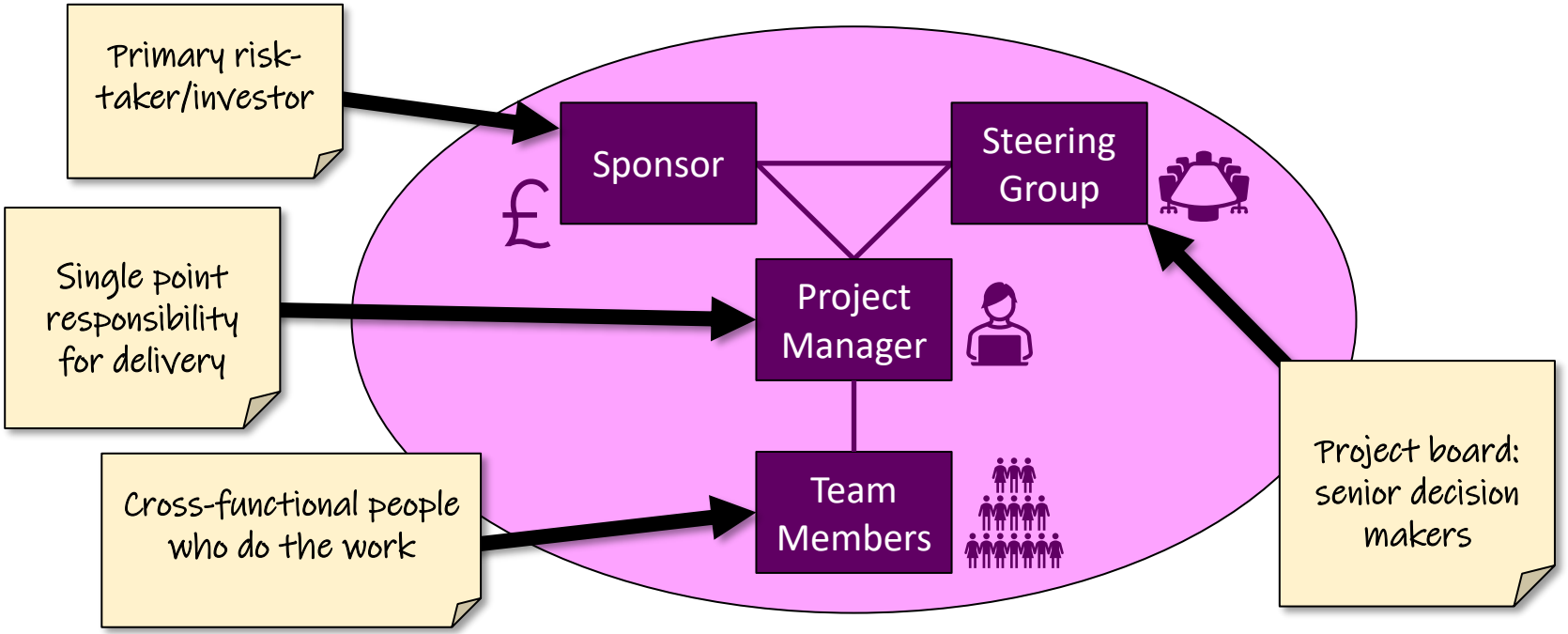
- **Who counts as a stakeholder?**
  - **End users** (the ones that use and benefit from the system)
  - **Managers and clients** (sponsors, strategic decision makers)
  - **Project build team** (developers, designers, testers, and project managers)
  - **External groups** (regulatory bodies, legal experts, customers, suppliers, shareholders, and community representatives)
  - **Support and maintenance staff**

Involving the right stakeholders from the start leads to clearer requirements, better software, and smoother project progress.

Stakeholders

Aston University
BIRMINGHAM UK

Representing stakeholders whilst ensuring a single-point responsibility for delivery



Setting clear project objectives

- Objectives define ***what*** *must be achieved* — not the detailed tasks. They focus on the *outcome*, not the *activity*.

- Focus on <u>what</u> will be put in place, rather than <u>how</u> activities will be carried out
  - **YES** Deliver an online booking system that allows users to register, make appointments, and receive email confirmations will be live by 1st December.
  - **NO** Design and code a booking website using HTML and JavaScript.

Objectives should be SMART

Example: Deliver an online booking system that allows users to register, make appointments, and receive email confirmations will be live by 1st December.

S – **Specific**: concrete and well-defined

M – **Measurable**, that is, satisfaction of the objective can be objectively judged

A – **Achievable**: it is within the power of the individual or group concerned to meet the target

R – **Relevant**: the objective must be relevant to the true purpose of the project

T – **Time-constrained**: time-bound, there is a defined point in time by which the objective should be achieved

Goals of Software Project Management

Aston University
BIRMINGHAM UK

- Given a set of SMART objectives, we want:
  - **Q** – quality software product
  - **C** – cost: no significant budget overshoot
  - **T** – time: no significant delays
  - **P** – good productivity

- Questions:
  - What is ‘software quality’ and (how) can it be measured?
  - Can we determine in advance what a project is going to cost?
  - How do we know if a significant deadline overshoot is likely?
  - Is ‘productivity’ measurable in a software engineering environment?

Levels of project objectives

- **Level 1: Operational**: activity level (short-term) Day-to-day activities which vary greatly depending on the project's process

- **Level 2: Tactical**: project level (mid-term) Quality, Time, Cost, Productivity

- **Level 3: Strategic**: organisational level (long-term)
  - reducing costs
  - expanding customer base
  - increasing user satisfaction
  - improving productivity
  - reducing time-to-market

Tactical project questions (examples)

- How should we ‘choose between competing designs’ (e.g., by complexity)?
- How long will this ‘phase of the project’ take?
- Are the ‘current project risks’ worth addressing? Do we really need to worry about them?
- Which ‘modules’ need extra attention during testing?
- Has ‘team productivity’ changed over the last few months?

(Project-Specific Issues)

Strategic project questions (examples)

Aston University
BIRMINGHAM UK

- Why are projects in this department often '*over schedule by 30%'* or more?
- Can we '*reduce future costs'* by consistently using this new software tool?
- Does '*Team X consistently produce better code than Team Y"*? If so, why?
- Would adopting a '*new team structure'* significantly improve time-to-market for in-house projects?
- Has our '*cost estimation model'* (used over the last 18 months) been accurate? Can it be calibrated for our environment?

(Long-Term, Organisation-Wide Issues)

Components of Software Project Management*

Aston University
BIRMINGHAM UK

- Planning – deciding what is to be done
- Organising – making arrangements
- Staffing – selecting the right people for the job
- Directing – giving instructions
- Monitoring – checking on progress
- Controlling – taking action to remedy hold-ups
- Innovating – coming up with solutions when problems emerge
- Representing – liaising with clients, users, developers and other stakeholders

**Adapted from D. Ince et al, Introduction to Software Project Management and Quality Assurance, McGraw Hill, 1993.*

Summary of key points in this unit

Aston University
BIRMINGHAM UK

- **Software projects differ significantly from other engineering projects**: new field, rapid change, high failure rate & complexity, no physical artefacts

- We need principled, quantitative approaches that can support informed management decisions at strategic, tactic and operational level

- **Specific – Measurable – Achievable – Relevant – Time-constrained SMART** project objectives are required to build quality software within budget, without delays and with good productivity

Review Questions

Aston University
BIRMINGHAM UK

- EXPLAIN WHY should the objectives of a software project be SMART? What does SMART stand for?
- EXPLAIN WHY is software project management important? Why is it needed?
- EXPLAIN the differences between a job, a project, and outright exploration.
- EXPLAIN WHY is software project management differs from other project management disciplines.

- **Start devising SMART objectives for your FYP** ✅

Next lectures (Weeks 2-3)

- Unit 2: Measurement, Estimation and Data Analysis

Aston University
BIRMINGHAM UK

Software Project Management

Unit 1: Introduction

Thais Webber
Richard Lee