

CSIT214/CSIT883
IT Project Management



Project procurement management

Project management framework (review)

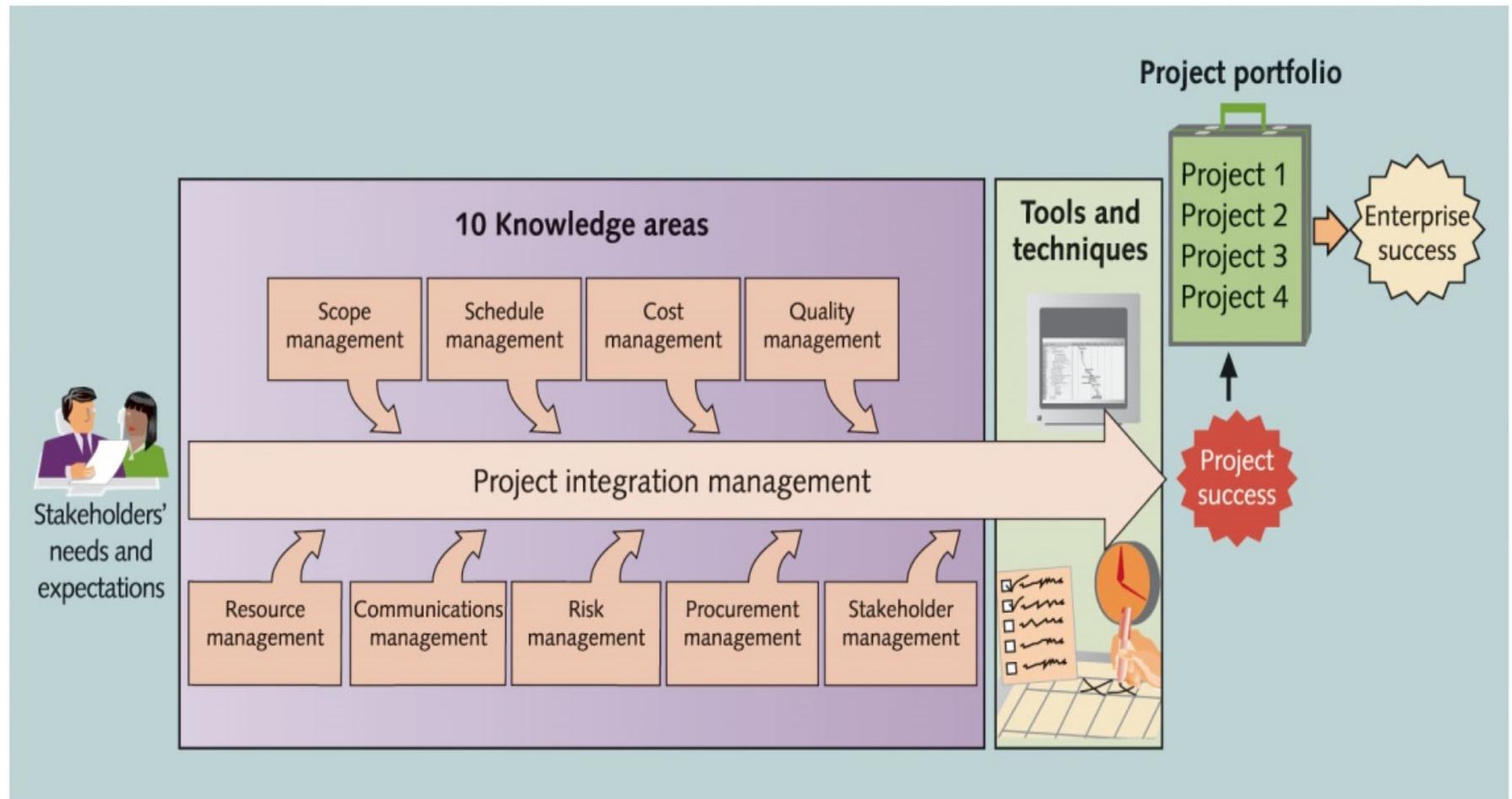


FIGURE 1-2 Project management framework

Contract



- ❑ Contracting is an integral part of doing business.
- ❑ Contracts can range significantly in value, in duration, and in complexity.

Acquiring software from external supplier

This could be:

- ❑ a *bespoke system* - created specially for the customer
- ❑ *off-the-shelf* - bought 'as is'
- ❑ *customised off-the-shelf* (COTS) - a core system is customised to meet needs of a particular customer

Goods vs. services vs. licenses

□ Goods

- E.g. Purchasing a car or a laptop

□ Services

- E.g. Fixing a window or writing a software

□ Licenses

- Permission to use the software
- The software still remains in the ownership of the supplier
 - The supplier may be able to grant the license to other customers.
- Often license to use software is bought rather than the software itself

Types of contract

- ❑ fixed price contracts
- ❑ time and materials contracts
- ❑ fixed price per delivered unit

Fixed price contracts

- ❑ **Price** is fixed when the contract is signed.
- ❑ Customer's **requirements** has to be fixed.
 - Detailed requirements analysis must have been done.
 - Requirements cannot be changed without renegotiating the price of the contract.
- ❑ Advantages
 - known expenditure
 - supplier motivated to be cost-effective
- ❑ Disadvantages
 - supplier will increase price to meet contingencies
 - cost of changes likely to be higher
 - threat to system quality

Time and materials

- ❑ Customer is charged at a fixed rate per unit of effort (e.g. per person-hour).
- ❑ Supplier invoices the customer for work done at regular intervals (e.g. each month).
- ❑ Advantages
 - easy to change requirements
 - lack of price pressure can assist product quality
- ❑ Disadvantages
 - Customer liability - the customer absorbs all the risk associated with poorly defined or changing requirements
 - Lack of incentive for supplier to be cost-effective

Fixed price per unit delivered

- Supplier gives a fixed-price per unit quote
- The size of the system is estimated
- The final price is the unit price multiplied by the number of (estimated) units.

<i>FP count</i>	<i>Design cost/FP</i>	<i>Implementation cost/FP</i>	<i>Total cost/FP</i>
Up to 2,000	\$242	\$725	\$967
2,001- 2,500	\$255	\$764	\$1,019
2,501-3,000	\$265	\$793	\$1,058
3,001-3,500	\$274	\$820	\$1,094
3,501-4,000	\$284	\$850	\$1,134

These figures do come from a real source (RDI Technologies in the USA).

Fixed price per unit delivered

<i>FP count</i>	<i>Design cost/FP</i>	<i>Implementation cost/FP</i>	<i>Total cost/FP</i>
Up to 2,000	\$242	\$725	\$967
2,001- 2,500	\$255	\$764	\$1,019
2,501-3,000	\$265	\$793	\$1,058
3,001-3,500	\$274	\$820	\$1,094
3,501-4,000	\$284	\$850	\$1,134

- Estimated system size 2,600 FPs
- Price
 - 2000 FPs x \$967 *plus*
 - 500 FPs x \$1,019 *plus*
 - 100 FPs x \$1,058
 - i.e. \$2,549,300
- What would be charge for 3,200 FPs?

Fixed price per unit

- How to estimate the construction effort needed when the requirements are **not** yet established?
 - One approach is to negotiate a **series of contracts**, each covering a different stage of development
 - Or, first carry out system design: a charge could be made. Then, if the design was then **implemented**, additional charge.

<i>FP count</i>	<i>Design cost/FP</i>	<i>Implementation cost/FP</i>	<i>Total cost/FP</i>
to 2,000	\$242	\$725	\$967
2,001- 2,500	\$255	\$764	\$1,019
2,501-3,000	\$265	\$793	\$1,058
3,001-3,500	\$274	\$820	\$1,094
3,501-4,000	\$284	\$850	\$1,134

Fixed price per unit

□ Advantages for customer

- customer understanding of how price is calculated
- comparability between different pricing schedules
- emerging functionality can be accounted for
- supplier incentive to be cost-effective

□ Disadvantages

- difficulties with software size measurement – may favour either developer or customer.
 - may need independent FP counter
- changing (as opposed to new) requirements: how do you charge?
 - Some requested changes may affect existing code drastically but not increase overall FP count
 - A suggestion: vary the charge depending on the time when change is requested.

Fixed price/unit

Table from draft Acquisition of Customised Software Policy document, published by Department of State Development, Victoria, Australia, 1996

	PRE-Acceptance Testing Handover	POST-Acceptance Testing Handover
Additional FPs	100%	100%
Changed FPs	130%	150%
Deleted FPs	25%	50%

Exercise: *A contract stipulates that a computer application is to be designed, constructed and delivered at a cost of \$600 per FP. After acceptance testing, the customer asks for changes to some of the functions in the system amounting to 500FPs and some new functions which amount to 200 additional FPs.*

Using the table above, calculate the additional charge.

Types of contract

A different classification

- Contracts are classified based on the approach that is used in contract selection:
 - Open
 - Restricted
 - Negotiated

The tendering process

- Open tendering
 - any supplier can bid in response to the *invitation to tender*
 - all tenders must be evaluated in the same way
 - government bodies may have to do this by local/international law (e.g. EU and World Trade Organization requirements)
 - To ensure public bodies do not unfairly favour local businesses.

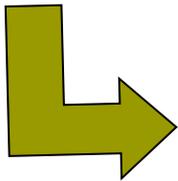
The tendering process

- Restricted tendering process
 - bids only from those specifically invited
 - can reduce suppliers being considered at any stage
- Negotiated procedure
 - negotiate with one supplier e.g. for extensions to software already supplied

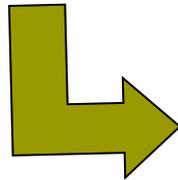
Stages in contract placement

requirements
analysis

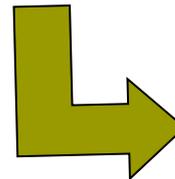
The **lack of, or errors in, the requirements** specification are probably the heart of most disputes resulting from the acquisition of computer equipment and software.



evaluation
plan



invitation to
tender



evaluation of
proposals

Requirements

A requirement is:

An informal (and/or formal) description or statement of a function, feature or condition that a user seeks to have implemented in a system.

There are 2 broad categories of requirements:

- Functional Requirements:
 - are those that relate directly to the functioning of the system.
- Non-functional requirements
 - cover aspects of the system such as user interface, performance, quality issues, interfaces to other systems, security etc.

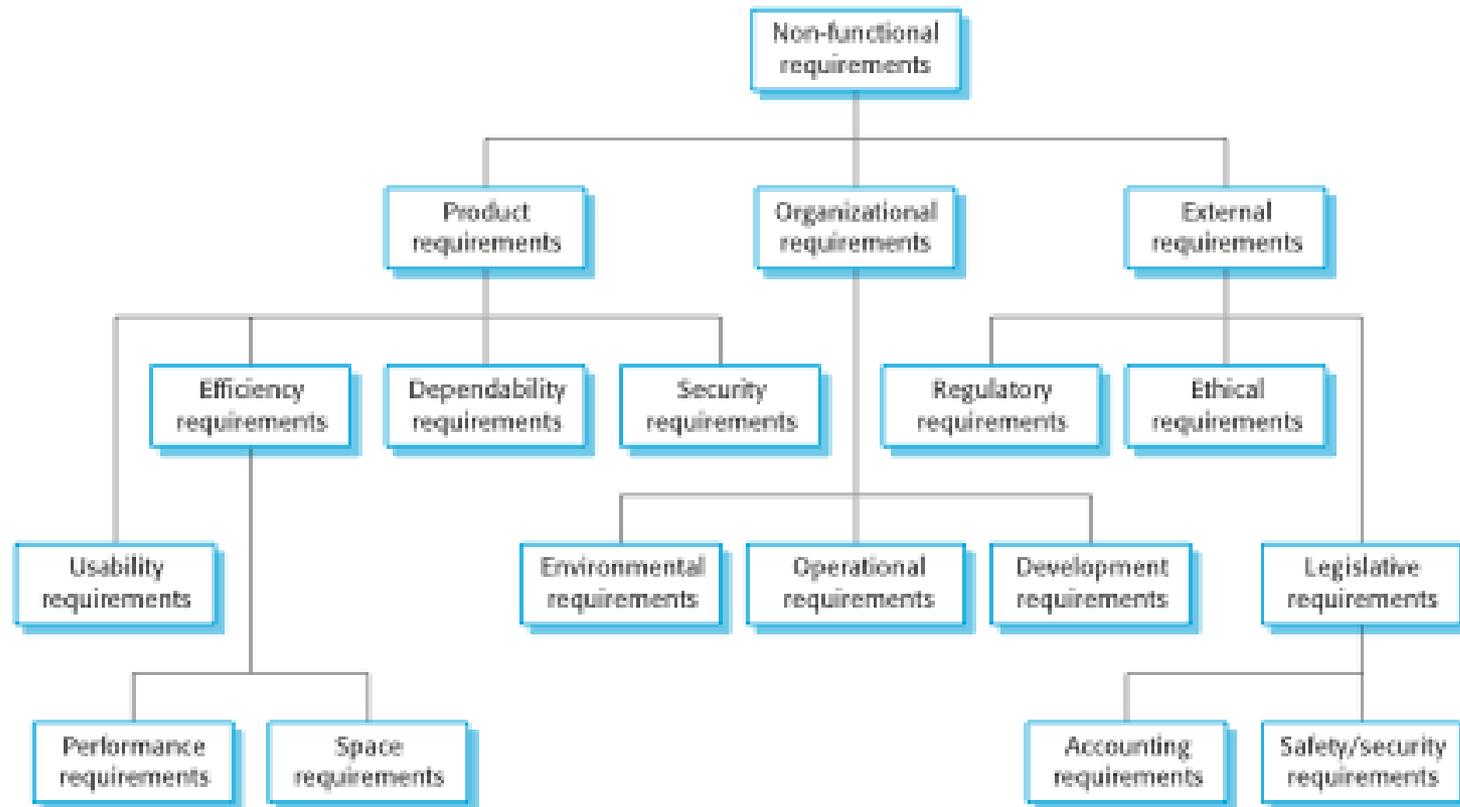
Functional requirements

- These specify what the system will do
 - The system should authenticate users before allowing them access.
 - The system should allow a user to check their bank account balance.
 - The system shall generate each day, for each clinic, a list of patients who are expected to attend appointments that day.

Non-functional requirements

- ❑ These are requirements **not directly related to system functions** for example performance requirements
- ❑ These are often very badly written in an ambiguous and totally untestable fashion e.g. “The system should be user friendly”
- ❑ We need to make them MEASURABLE

Types of nonfunctional requirement



Examples of nonfunctional requirements

Product requirement

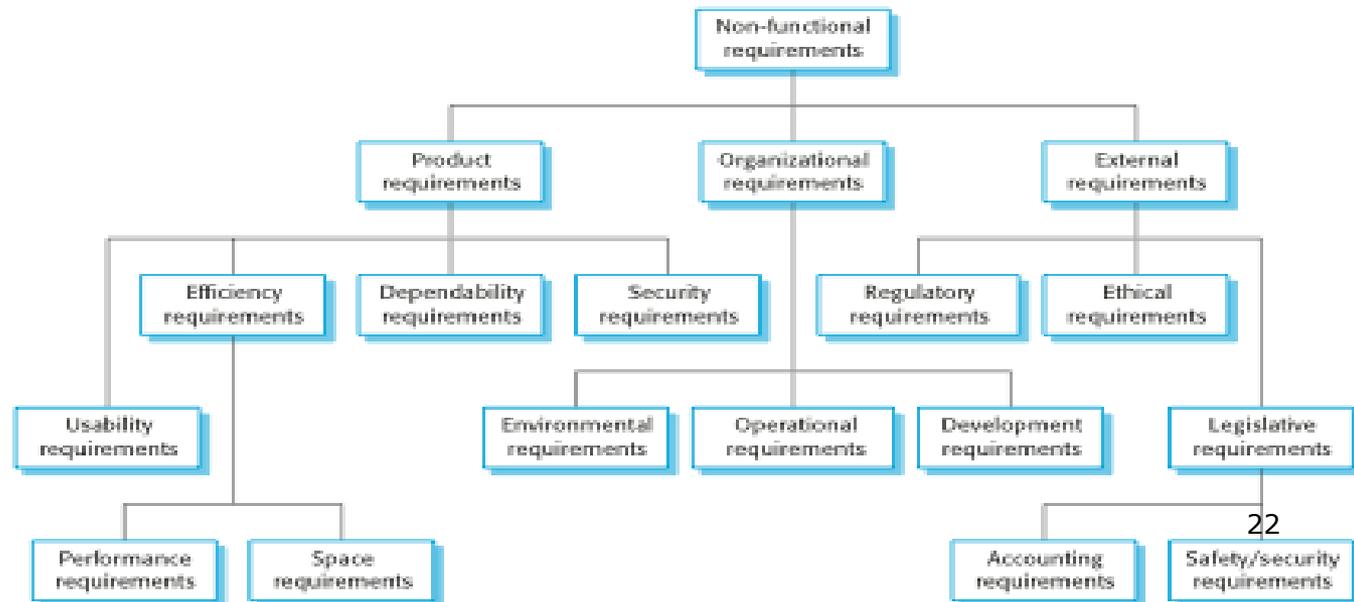
The MHC-PMS shall be available to all clinics during normal working hours (Mon–Fri, 0830–17.30). Downtime within normal working hours shall not exceed five seconds in any one day.

Organizational requirement

Users of the MHC-PMS system shall authenticate themselves using their health authority identity card.

External requirement

The system shall implement patient privacy provisions as set out in HStan-03-2006-priv.



Defining non-functional requirements

Here is a checklist of measurable properties.

Property

Speed

Metric

Processed transactions/sec
User/Event response time
Screen refresh time

Size

Kbytes
Memory

Ease of use

Training Time
Number of help frames

Reliability

Mean time to failure
Probability of unavailability
Rate of failure occurrence
Availability
Time to restart after failure
% of events causing failure

Portability

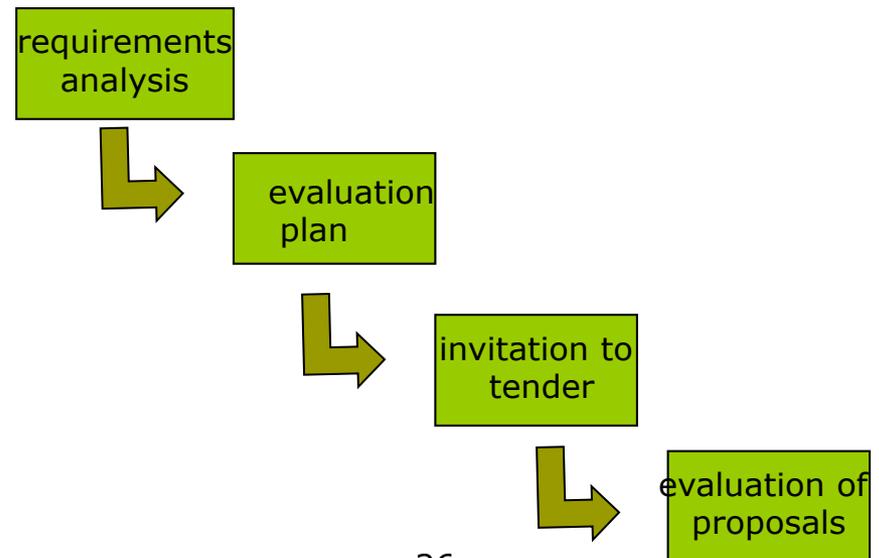
No. of target systems

Some other examples of non-functional requirements

- ❑ "The system should be easy to use by experienced controllers and should be organised in such a way that user errors are minimised"
= poor specification - not able to be tested
- ❑ "Experienced controllers should be able to use all of the system function after a total of 2 hours training. After this training, the average number of errors made by experienced users should not exceed 2 per day"
= GOOD specification - able to be tested.

Evaluation plan

- How are proposals to be evaluated?
- Methods could include:
 - reading proposals
 - interviews
 - demonstrations
 - site visits
 - practical tests



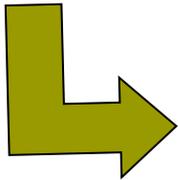
Evaluation plan -contd.

- ❑ Need to assess value for money (VFM) for each desirable feature
- ❑ Example:
 - Automated system to input data (replacing manual data entry a hired staff)
 - 4 hours work a month saved at \$20 an hour
 - The system to be used for 4 years
 - If the cost of feature is \$1000, would it be worth it?

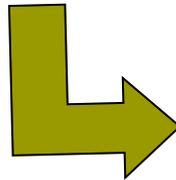
Stages in contract placement

requirements
analysis

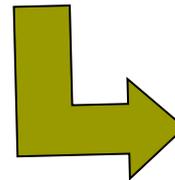
The lack of, or errors in, the requirements specification are probably the heart of most disputes resulting from the acquisition of computer equipment and software.



evaluation
plan



invitation to
tender



evaluation of
proposals

Invitation to tender (ITT)

- ❑ Note that bidder is making an *offer* in response to ITT
- ❑ *Acceptance* of offer creates a *Contract*
- ❑ Customer may need further information
- ❑ Problem of different technical solutions to the same problem

ISO 12207 refers to an ITT as a Request for Proposal (RFP).

Tendering process for complex projects

- Technical proposals are requested from **potential suppliers**.
- Some of these proposals can be **dismissed** as not meeting mandatory requirements.
- Remaining ones could be discussed with representatives of suppliers to **clarify and validate** technical proposals
- Suppliers might be asked to **demonstrate** certain aspects of their proposals
- Supplier could be given opportunity to **remedy shortcomings** detected.

Two-stage tendering process for complex projects

- These discussions could result in a **Memorandum of Agreement (MoA)** with each prospective supplier
 - This is an acceptance by the customer that the proposed solution satisfies customer's requirement

Contract checklist

- **Definitions** – what words mean precisely e.g. 'supplier', 'user', 'application'
- **Form of agreement.** For example, is this a contract for a sale or a lease, or a license to use a software application? Can the license be transferred?
- **Goods and services to be supplied** – this could include lengthy specifications
- **Timetable** of activities
- **Payment arrangements** – payments may be tied to completion of specific tasks

Contract checklist - continued

□ **Ownership of software**

- Can client sell software to others?
- Can supplier sell software to others? Could specify that customer has 'exclusive use'
- Does supplier retain the copyright?
- Where supplier retains source code, may be a problem if supplier goes out of business; to circumvent a copy of code could be deposited with an escrow service

Contract management

Some terms of contract will relate to management of contract, for example,

- Progress reporting
- Decision points – could be linked to release of payments to the contractor
- Variations to the contract, i.e. how are changes to requirements dealt with?
- Acceptance criteria