

SE345

Atilim University
Dept of Software Engineering

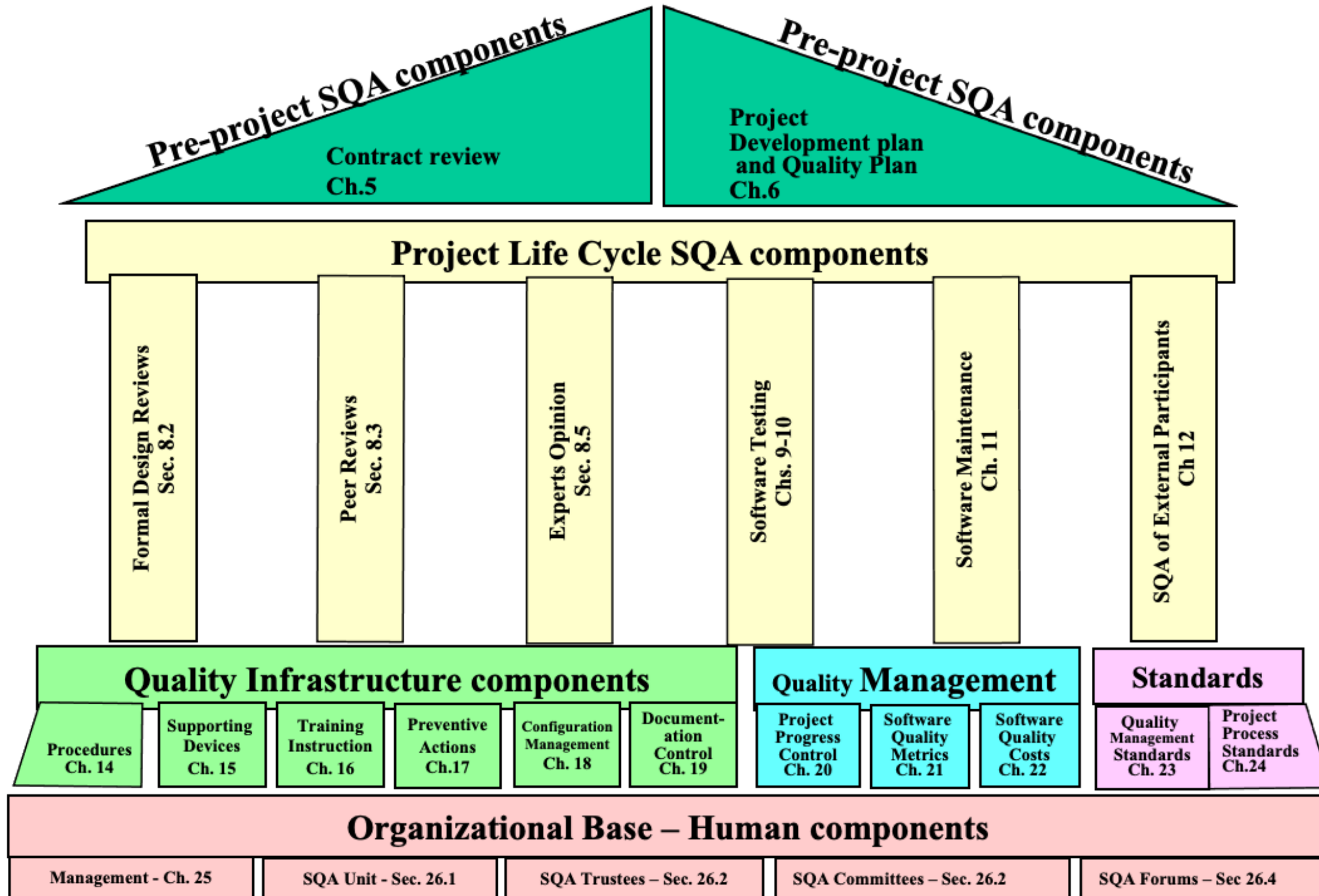
Asst. Prof. Dr. Aylin AKCA-OKAN



Tentative Course Schedule

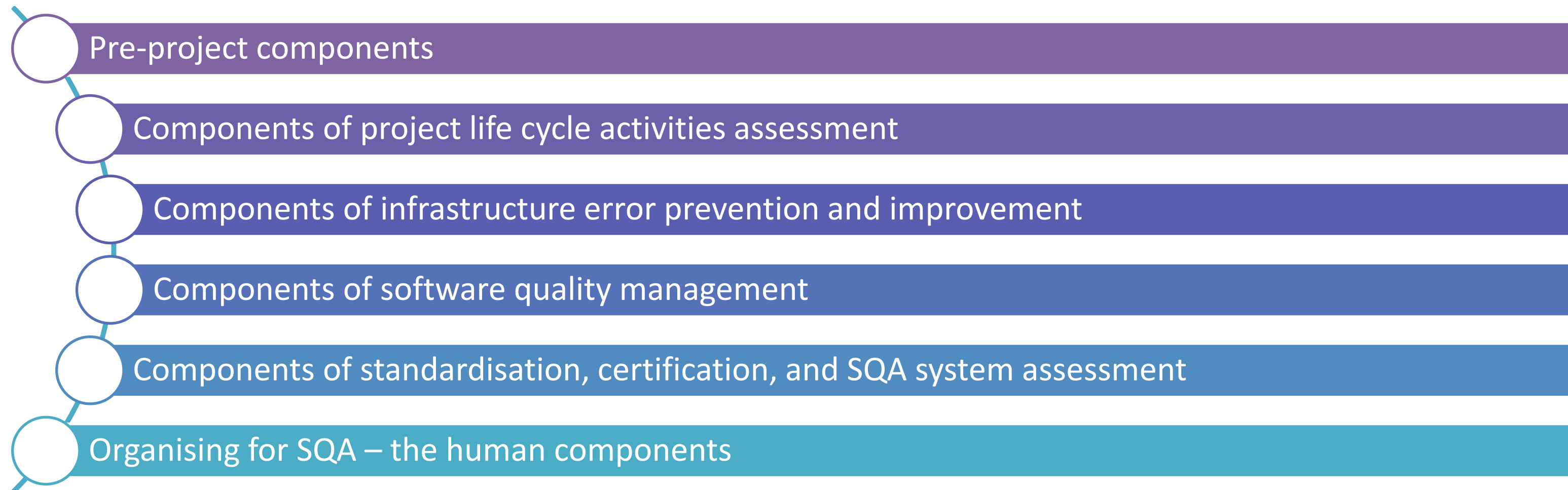
Wk	Subjects	Chapter
1	Introduction to Software Quality and Assurance	Chapter 1
2	Introduction to Software Quality and Assurance	Chapter 1
3	Software Quality Factors	Chapter 3
4	Overview of Components of the SQA System	Chapter 4
5	Overview of Components of the SQA System	Chapter 4
6	Presentation	Chapter 9
7	Midterm	
8	Integrating Quality Activities in Project Life Cycle	Chapter 7
9	Software Quality Metrics	Chapter 21
10	Reviews, Inspection and Audits	Chapter 8
11	Procedures and Work Instructions	Chapter 14
12	In-class Project	
13	Software Change Process	Chapter 18
14	SOA Process Standards	Chapter 23

The Software Quality Shrine



The SQA system - An SQA architecture

SQA system components can be classified into six classes:



1. Pre-project components

To ensure that

- The project commitments have been adequately defined considering the resources required, the schedule and budget.
- The development and quality plans have been correctly determined.

to improve the preparatory steps taken prior to initiating work on the project itself

Contract reviews

Development and quality plans

Contract review

Accordingly, contract review activities must include a detailed examination of

the project proposal draft

the contract drafts

Specifically, contract review activities include:

- Clarification of the customer's requirements
- Review of the project's schedule and resource requirement estimates
- Evaluation of the professional staff's capacity to carry out the proposed project
- Evaluation of the customer's capacity to fulfill his obligations
- Evaluation of development risks.

Pre-project components

Development and Quality Plan

Development Plan

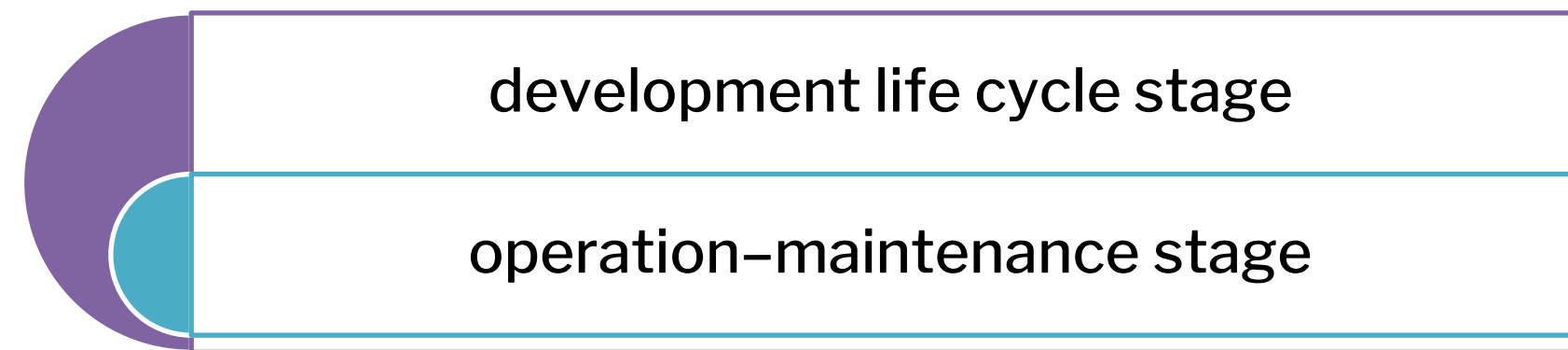
- ◎ Schedules
- ◎ Required manpower and hardware resources
- ◎ Risk evaluation
- ◎ Organizational issues: team members, subcontractors and partnerships
- ◎ Project methodology, development tools etc.
- ◎ Software reuse plans

Quality Plan

- ◎ Quality goals, expressed in the appropriate measurable terms
- ◎ Criteria for starting and ending each project stage
- ◎ List of reviews, tests and other scheduled verification and validation activities

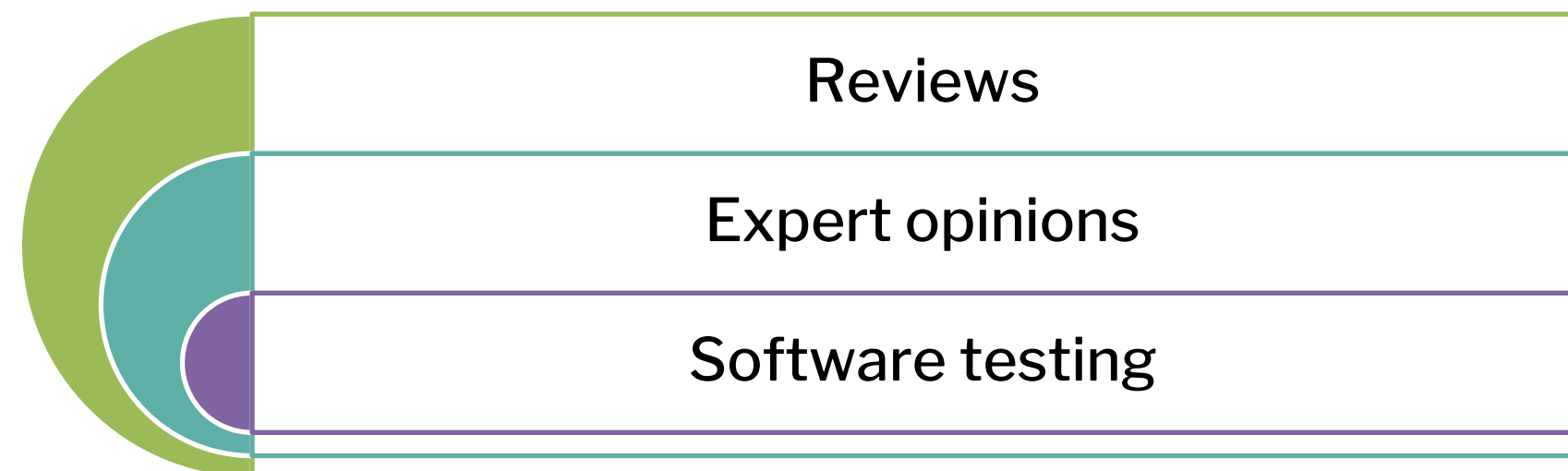
2. Components of project life cycle activities assessment

The project life cycle is composed of two stages:



Development life cycle activities assessment

The development life cycle stage components detect design and programming errors. Its components are divided into the following three sub-classes:



Development life cycle components

Reviews

Software development process produces a variety of documents.

Developers can only continue to the next phase of the development process **only on receipt of formal approval of these documents.**

A significant portion of these documents requires formal **professional approval** of their quality as required in the development contract and demanded by the procedures applied by the software developer.

Development life cycle components

Expert opinions

Useful in these situations

support by introducing additional external capabilities into the organization's in-house development process.

- Insufficient in-house professional capabilities in a given area.
- Temporary inaccessibility of in-house professionals(waiting will cause substantial delays in the project completion schedule).
- In cases of major disagreement among the organization's senior professionals, an outside expert may support a decision.

Development life cycle components

Expert opinions

Useful in these situations

support by introducing additional external capabilities into the organization's in-house development process.

In small organizations

- in many cases it is difficult to find enough suitable candidates to participate in the design review (DR) teams. In such situations, outside experts may join a DR committee or, alternatively, their expert opinions may replace a DR.
- Characterised by extreme work pressures, an outside expert's opinion can replace an inspection.

Development life cycle components

Testing

Software tests

```
graph TD; A[Software tests] -.-> B[formal SQA components that are targeted toward review of the actual running of the software.]; A -.-> C[based on a prepared list of test cases that represent a variety of expected scenarios]; A -.-> D[examine software modules (unit), software integration (integration), or entire software packages (systems). Recurrent tests (usually termed “regression tests”), carried out after correction of previous test findings, are continued till satisfactory results are obtained];
```

formal SQA components that are targeted toward review of the actual running of the software.

based on a prepared list of test cases that represent a variety of expected scenarios

examine software modules (unit), software integration (integration), or entire software packages (systems). Recurrent tests (usually termed “regression tests”), carried out after correction of previous test findings, are continued till satisfactory results are obtained

Maintenance life cycle components

The SQA components used during the operation -maintenance phase include specialised maintenance components as well as development life cycle components, which are applied mainly for functionality improving maintenance tasks.

An additional sub-class of SQA project life cycle components deals with assuring the quality of project parts performed by subcontractors and other external participants during project development and maintenance.



Software Maintenance Components

Maintenance Services

```
graph TD; MS[Maintenance Services] -.-> SQA[involves the application of a great variety of SQA components (similar to development phase)]; MS -.-> Duration[vary in range and are provided for extensive periods, often several years]; MS -.-> Quality[should meet all kinds of quality requirements, particularly functionality and scheduling requirements (generally decided together with the customer) as well as budget limitations (determined by the service provider)];
```

involves the application of a great variety of SQA components (similar to development phase)

vary in range and are provided for extensive periods, often several years

should meet all kinds of quality requirements, particularly functionality and scheduling requirements (generally decided together with the customer) as well as budget limitations (determined by the service provider)

Maintenance life cycle components

Assurance of the quality of external participants' work (e.g. Subcontractors)

Why working with external participants?

economic, technical, personnel related interests/reasons

Clients + Subcontractors +
Developers (suppliers)

The larger and more complex the project

The greater the likelihood that external participants will be required, and the larger the proportion of work transmitted to them (subcontractors, suppliers of COTS software and the customer)

Software lifecycle components

Assurance of the quality of external participants' work (e.g. Subcontractors)

RISK

Not meeting schedule or other requirements, if an external participant's work is performed using software assurance standards below those of the supplier's

Effective controls needed over the external participant's work

3. Components of infrastructure error prevention and improvement

The main objectives of these components, which are applied throughout the entire organisation, are to eliminate or at least reduce the rate of errors, based on the organisation's accumulated SQA experience.

Infrastructure components for error prevention and improvements

Aim

- ◎ to eliminate or at least reduce the rate of errors
- ◎ to prevent software faults or, at least, the lowering of software fault rates, together with the improvement of productivity

- Procedures and work instruction
- Templates and checklists
- Staff training, retraining and certification
- Preventive and corrective actions
- Configuration management
- Documentation control

Infrastructure components for error prevention and improvements

Procedures and work instruction



planned to be generally applicable and to serve the entire organization

provide detailed directions for the use of methods that are applied in unique instances and employed by specialized teams

Infrastructure components for error prevention and improvements

Templates and Checklists

Save the time required to define the structure of the various documents or prepare lists of subjects to be reviewed

Contribute to the completeness of the documents and reviews

Improve communication between development team and review committee members by standardising documents and agendas

Infrastructure components for error prevention and improvements

Configuration management

The regular software development and maintenance operations involve:

- intensive activities that modify software to create new versions and releases.

These activities are conducted throughout the entire software service period in order to cope with the needed corrections, adaptations to specific customer requirements, application improvements, and so forth.

Different team members carry out these activities simultaneously, although they may take place at different sites. As a result, serious dangers arise, whether of misidentification of the versions or releases, loss of the records delineating the changes implemented, or loss of documentation.

➤ Consequently failures may be caused.

Infrastructure components for error prevention and improvements

Configuration management

Configuration management deals with these hazards by introducing procedures to control the change process.

These procedures relate to the approval of changes, the recording of those changes performed, the issuing of new software versions and releases, the recording of the version and release specifications of the software installed in each site, and the prevention of any changes in approved versions and releases once they are issued.

Most configuration management systems implement computerised tools to accomplish their tasks.

These computerised systems provide the updated and proper versions of the installed software for purposes of further development or correction. Software configuration procedures generally authorise an administrator or a configuration management committee to manage all the required configuration management operations.

Infrastructure components for error prevention and improvements

Configuration Management

Development & maintenance ops : intensive activities that **modify software** to create new versions and releases

Procedures to control the change process!

- approval of changes
- recording of those changes performed
- issuing new software versions and releases
- recording of the version and release specifications of the software installed in each site
- prevention of any changes in approved versions and releases once they are issued

Infrastructure components for error prevention and improvements

Documentation control

SQA requires the application of measures to ensure the efficient long-term availability of primary documents related to software development (“controlled documents”).

The purpose of one type of controlled document – the quality record – is mainly to provide evidence of the SQA system’s performance.

Documentation control, therefore, represents one of the building blocks of any SQA system.

Infrastructure components for error prevention and improvements

Documentation control

- Definition of the types of controlled documents needed
- Specification of the formats, document identification methods, etc.
- Definition of review and approval processes for each controlled document
- Definition of the archive storage methods.
- Controlled documents contain information important to the long-term development and maintenance of the software system, such as software test results, design review (DR) reports, problem reports, and audit reports.
- Quality records mainly contribute to the system's ability to respond to customer claims in the future.

Infrastructure components for error prevention and improvements

Staff training, retraining and certification

Key to efficient, quality performance!

Keep staff knowledgeable and updated by

- Training new employees and retraining those employees who have changed assignments
- Continuously updating staff with respect to professional developments and the in-house, hands-on experience acquired
- Certifying employees after their knowledge and ability have been demonstrated

Infrastructure components for error prevention and improvements

Preventive and corrective actions

Systematic study of the data collected regarding instances of failure and success

Some of the sources: design review reports, software test reports, and customers' complaints

- Implementation of changes that prevent similar failures in the future
- Correction of similar faults found in other projects and among the activities performed by other teams
- Implementing proven successful methodologies to enhance the probability of repeat successes

4. Components of software quality management

This class of components is geared toward several goals, the major ones being the control of development and maintenance activities and the introduction of early managerial support actions that mainly prevent or minimize schedule and budget failures and their outcomes.

Management SQA components

- Managerial SQA components support the managerial control of software development projects and maintenance services.

Control components include:

- Project progress control (including maintenance contract control)
- Software quality metrics
- Software quality costs.

Project progress control

The main objective of project progress control components is to detect the appearance of any situation that may induce deviations from the project's plans and maintenance service performance.

Project control activities focus on:

- Resource usage
- Schedules
- Risk management activities
- The budget.

Software quality metrics

Measurement of the various aspects of software quality is considered to be an effective tool for the support of control activities and the initiation of process improvements during the development and the maintenance phases.

We can list metrics for:

- Quality of software development and maintenance activities
- Development teams' productivity
- Help desk and maintenance teams' productivity
- Software faults density
- Schedule deviations
- Components

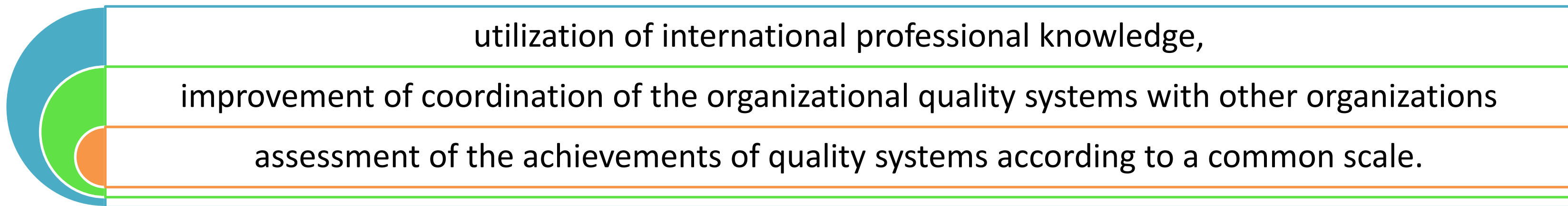
Software quality costs

The quality costs incurred by software development and application are, according to the extended quality costs model,

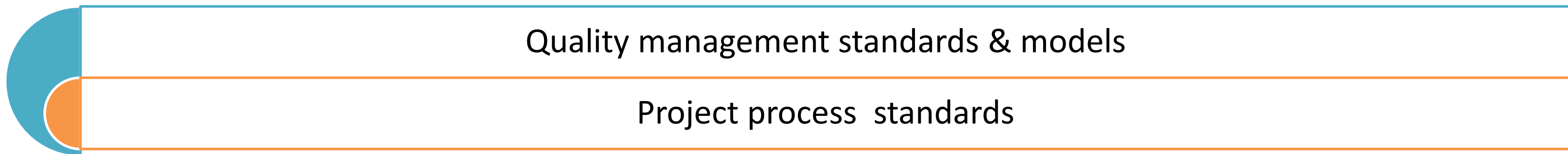
- the costs of control (prevention costs, appraisal costs, and managerial preparation and control costs)
- combined with the costs of failure (internal failure costs, external failure costs, and managerial failure costs).

5. Components of standardisation, certification, and SQA system assessment

**These components implement international professional and managerial standards within the organization.
The main objectives of this class are :**



The various standards may be classified into two main groups:



SQA standards, system certification, and assessment components

External tools offer another avenue for achieving the goals of software quality assurance.

Specifically, the main objectives of this class of components are:

- Utilisation of international professional knowledge.
- Improvement of coordination with other organisations' quality systems.
- Objective professional evaluation and measurement of the achievements of the organisation's quality systems

The standards available may be classified into two main subclasses:

- quality management standards (or models) and
- project process standards.

Either or both of the two subclasses can be required by the customer and stipulated in the accompanying contractual agreements.

Quality management standards & models

The organisation can clearly benefit from quality standards of the second subclass that guide the management of software development, maintenance, SQA standards, system certification, and assessment components and infrastructure.

These standards focus on *what* is required and leave the decision about *how* to achieve it to the organisation.

The application of a managerial quality system provides a fairly objective assessment of the organisation's achievements.

Organisations that comply with quality achievement requirements can then seek SQA certification.

The most familiar examples of this type of standard and models are:

- SEI CMMI assessment models
- ISO 9001 and ISO 9000-3 standards
- AQAP (Allied Quality Assurance Publications)
- ISO /IEC 15504 – (SPICE model - Software Process Improvement and Capability Determination)

Project process standards

Project process standards are professional standards that provide methodological guidelines (dealing with the question of “how”) for the development team.

Well-known examples of this type of standards are:

- ISO/IEC 12207 (Standard for Information Technology– Software Life Cycle Processes)
- IEEE 1012 (Standard for Software Verification and Validation)

6. Organizing for SQA – the human components

The SQA organisational base includes managers, testing personnel, the SQA unit and practitioners interested in software quality (SQA trustees, SQA committee members and SQA forum members).

All these *actors* contribute to software quality;

- **their main objectives are to initiate and support the implementation of SQA components, detect deviations from SQA procedures and methodology, and suggest improvements.**

Organizing for SQA - the Human component

Main objectives

- To develop and support implementation of SQA components
- To detect deviations from SQA procedures and methodology
- To suggest improvements to SQA components

- Management's role in SQA
- The SQA unit
- SQA trustees
- SQA committees
- SQA forums

Organizing for SQA - the Human component

Management's role in SQA

The responsibilities of

- top management
- departmental management and
- project management

- Definition of the quality policy
- Effective follow-up of quality policy implementation
- Allocation of sufficient resources to implement quality policy
- Assignment of adequate staff
- Follow-up of compliance of quality assurance procedures
- Solutions of schedule, budget and customer relations difficulties.

Organizing for SQA - the Human component

The SQA unit

The SQA unit + testers

devote themselves to SQA activities

- Preparation of annual quality programs
- Consultation with in-house staff and outside experts on software quality issues
- Conduct of internal quality assurance audits
- Leadership of quality assurance various committees
- Support of existing quality assurance infrastructure components and their updates, and development of new components

Organizing for SQA - the Human component

The SQA Trusties

Members of development and maintenance teams who have a special interest in software quality

- Solving team or unit local quality problems
- Detecting deviations from quality procedures and instructions
- Initiating improvements in SQA components
- Reporting to the SQA unit about quality issues in their team or unit

Organizing for SQA - the Human component

The SQA Committees

members of various software
development and maintenance units

- Solution of software quality problems.
- Analysis of problem and failure records as well as other records, followed by initiation of corrective and preventive actions when appropriate.
- Initiation and development of new procedures and instructions; updating existing materials.
- Initiation and development of new SQA components and improvement of existing components

Organizing for SQA - the Human component

The SQA Forums

professionals and practitioners who meet and/or maintain an Internet site on a voluntary basis for discussion of quality issues pertaining to development and maintenance processes

share their experiences and difficulties as well as try to initiate improvements in the software process

important sources of information and SQA initiatives

Summary

SQA Area Name	Focus	Alignment with Project Life Cycle
Area 1: SQA Process Implementation	Establishing SQA infrastructure and planning (Chapter 7/Part II).	Preparing the SQAP, contract review support, SQA records control.
Area 2: Product Assurance Activities for Conformance	Evaluation and measurement of software products.	Evaluating products for conformance, assuring operation services quality, product quality metrics.
Area 3: Process Assurance Activities for Conformance	Evaluation and measurement of processes and improvements.	Evaluating processes for conformance, corrective/preventive actions (CAPA), subcontractor process assurance, process quality metrics, training/certification assessment.

SQAP Element Group	Focus	Examples of Tasks
SQA Process Implementation (Area 1)	Establishing infrastructure and planning SQA effort.	Preparing the SQAP, contract review support, documentation control, planning for correction of management deficiencies.
Product Assurance (Area 2)	Evaluating artifacts <i>for conformance</i> to requirements and standards.	Evaluation of project plans, evaluation of software products, product acceptability evaluation, software product quality metrics.
Process Assurance (Area 3)	Evaluating the development <i>processes</i> and environment <i>for conformance</i> .	Evaluation of life cycle processes, evaluation of environment, evaluation of subcontractors, process measurement/metrics, staff skills assessment.