12. ISO 9000 STANDARD

- ISO (International Standards Organization) is a consortium of 63 countries established to formulate and foster standardization. ISO published its 9000 series of standards in 1987. ISO certification serves as a reference for contract between independent parties.
- The ISO 9000 standard specifies the guidelines for maintaining a quality system. The ISO standard mainly addresses operational aspects and organizational aspects such as responsibilities, reporting, etc.
- In a nutshell, ISO 9000 specifies a set of guidelines for repeatable and high quality product development. ISO 9000 standard is a set of guidelines for the production process and is not directly concerned about the product itself.

- **OVERVIEW OF ISO 9000 SERIES**

ISO Standard	**Title**	**Description**
ISO 9000	*Quality Management and Quality Assurance Standards--Guidlines for Selection and Use*	Guidelines for the selection and use of ISO 9001, 9002 and 9003.
ISO 9001*	*Quality Systems--Model for quality assurance in design/development, production, installation and servicing.*	Standard covers design, development, production, installation, and servicing, **this applies to the software industry.**
ISO 9002	*Quality systems--Model for quality assurance in production and installation.*	Assesses the production and installation processes.
ISO 9003	*Quality systems--Model for quality assurance in final inspection and test.*	Evaluation the final inspection and test phase.
ISO 9004	*Quality management and quality system elements—Guidelines*	Defines the 20 fundamental quality system concepts included in the three models.

Need for obtaining ISO 9000 certification / Why to get ISO certification

- Confidence of customers in an organization increases when organization qualifies for ISO certification. This is especially true in the international market. In fact, many organizations awarding international software development contracts insist that the development organization have ISO 9000 certification. For this reason, it is vital for software organizations involved in software export to obtain ISO 9000 certification.
- ISO 9000 requires a well-documented software production process to be in place. A well-documented software production process contributes to repeatable and higher quality of the developed software.
- ISO 9000 makes the development process focused, efficient, and cost effective.
- ISO 9000 certification points out the weak points of an organization and recommends remedial action.
- ISO 9000 sets the basic framework for the development of an optimal process and Total Quality Management (TQM).

HOW TO GET ISO 9000 CERTIFICATION

1. **Application stage:** The organisation should apply for ISO certification to registrar for registration
2. **Pre –assessment:** During this stage the registrar makes a rough assessment of the organisation.
3. **Document review & adequacy audit:** The registrar reviews the documents submitted by the organisation and makes suggestions for possible improvements.
4. **Compliance audit:** The registrar checks whether the suggestions made by it during review have been complied to by the organisation or not.
5. **Registration:** The registrar awards the ISO 9000 certificate after successful completion of all previous phases.
6. **Continued surveillance:** The registrar continues monitoring the organisation periodically.

Summary of ISO 9001 certification

A summary of the main requirements of ISO 9001 as they relate of software development is as follows:

Management Responsibility (4.1)

- The management must have an effective quality policy.
- The responsibility and authority of all those whose work affects quality must be defined and documented.
- A management representative, independent of the development process, must be responsible for the quality system. This requirement probably has been put down so that the person responsible for the quality system can work in an unbiased manner.
- The effectiveness of the quality system must be periodically reviewed by audits.

Quality System (4.2)

A quality system must be maintained and documented.

Contract Reviews (4.3)

Before entering into a contract, an organization must review the contract to ensure that it is understood, and that the organization has the necessary capability for carrying out its obligations.

Design Control (4.4)

- The design process must be properly controlled, this includes controlling coding also. This requirement means that a good configuration control system must be in place.
- Design inputs must be verified as adequate.
- Design must be verified.
- Design output must be of required quality.
- Design changes must be controlled.

Document Control (4.5)

- There must be proper procedures for document approval, issue and removal.
- Document changes must be controlled. Thus, use of some configuration management tools is necessary.

Purchasing (4.6)
Purchasing material, including bought-in software must be checked for conforming to requirements.

Purchaser Supplied Product (4.7)
Material supplied by a purchaser, for example, client-provided software must be properly managed and checked.

Product Identification (4.8)
The product must be identifiable at all stages of the process. In software terms this means configuration management.

Process Control (4.9)

- The development must be properly managed.
- Quality requirement must be identified in a quality plan.

Inspection and Testing (4.10)
In software terms this requires effective testing i.e., unit testing, integration testing and system testing. Test records must be maintained.

Inspection, Measuring and Test Equipment (4.11)
If integration, measuring, and test equipment are used, they must be properly maintained and calibrated.

Inspection and Test Status (4.12)
The status of an item must be identified. In software terms this implies configuration management and release control.

Control of Nonconforming Product (4.13)
In software terms, this means keeping untested or faulty software out of the released product, or other places whether it might cause damage.

Corrective Action (4.14)
This requirement is both about correcting errors when found, and also investigating why the errors occurred and improving the process to prevent occurrences. If an error occurs despite the quality system, the system needs improvement.

Handling, (4.15)
This clause deals with the storage, packing, and delivery of the software product.

Quality records (4.16)
Recording the steps taken to control the quality of the process is essential in order to be able to confirm that they have actually taken place.

Quality Audits (4.17)
Audits of the quality system must be carried out to ensure that it is effective.

Training (4.18)
Training needs must be identified and met.

FEATURES OF ISO 9001 CERTIFICATION

The salient features of ISO 9001 are as follows:

- **Document control:** All documents concerned with the development of a software product should be properly managed, authorized, and controlled. This requires a configuration management system to be in place.
- **Planning:** Proper plans should be prepared and then progress against these plans should be monitored.
- **Review:** Important documents should be independently checked and reviewed for effectiveness and correctness.
- **Testing:** The product should be tested against specification.
- **Organizational aspects:** Several organizational aspects should be addressed e.g., management reporting of the quality team.

SHORTCOMINGS OF ISO 9000

- ISO 9000 requires a software production process to be adhered to but does not guarantee the process to be of high quality. It also does not give any guideline for defining an appropriate process.
- ISO 9000 certification process is not fool-proof and no international accreditation agency exists. Therefore it is likely that variations in the norms of awarding certificates can exist among the different accreditation agencies and also among the registrars.
- Organizations getting ISO 9000 certification often tend to downplay domain expertise. These organizations start to believe that since a good process is in place, any engineer is as effective as any other engineer in doing any particular activity relating to software development. However, many areas of software development are so specialized that special expertise and experience in these areas (domain expertise) is required. In manufacturing industry there is a clear link between process quality and product quality. Once a process is calibrated, it can be run again and again producing quality goods. In contrast, software development is a creative process and individual skills and experience are important.
- ISO 9000 does not automatically lead to continuous process improvement, i.e. does not automatically lead to TQM.

ISO 9000-3 Model

ISO 9000-3 divides the elements of its quality system model into three parts: a framework, the supporting structures, and the life cycle activities. The terminology of 9000-3 is software development-oriented, rather than manufacturing-oriented. Nevertheless, all of 9001’s requirements are covered by 9000-3.

ISO 9000-3, Primary Areas of Concern

The ISO 9000 Approach to Quality Systems for Software Development, describes some of the primary areas of concern in 9000-3. These include configuration management, change control, development planning, quality planning, design and implementation, testing and validation, and maintenance.

Configuration Management

The purpose of configuration management is to ensure each build of a product is derived from the correct version of every source file. These requirements apply to all software products, including customer documentation and manuals. A configuration management plan should identify the activities to be carried out and the tools, techniques, and methods to be used.

The plan should determine the stage at which items will be brought under configuration control. Each software item should be identified uniquely and each item's build status should be identified. The version of each software item required to construct each version of the product should be specified.

Change Control

Change control helps managers balance customer needs, technical capability, and additional resource requirements for modifications to the product. It helps prevent the late introduction of new requirements in the development cycle Change control also ensures all parties affected by a change are notified. Procedures should be developed to identify, document, review, and authorize any changes to items under configuration management.

It should be possible to trace approved changes and subsequent modifications to all affected items, from change initiation through release. Changes are often initiated by engineering change orders (ECOs) also known as modification requests, initiation requests, etc. A review board typically reviews the ECOs, prioritizes them, and decides whether to accept immediately, defer, or reject the requests based on previously established criteria.

Development Planning

A development plan provides a unified view of the activities required to complete a project. It includes identification of the activities to be performed, the resources required for each activity, and the timing of and coordination of each activity. Risk analysis and contingency planning should be included as part of the development planning process.

A development plan should define the development stages to be carried out for each project, the required inputs and outputs for each phase, and the verification procedures for each phase. The development plan should include a schedule, describe the mechanisms for tracking progress, and outline organizational responsibilities and work assignments. The development plan should identify related project plans such as the quality plan, integration plan, test plan, maintenance plan, and configuration management plan.

The development plan provides the basis for tracking project progress. Completion of milestones should be recorded and progress should be compared to the development plan schedule on a regular basis. Slippages should be identified and the plan should be updated to accommodate additional resources required, including time, personnel, and equipment. If the shipping date will be changed, affected departments and customers should be notified so they may plan accordingly.

Quality Planning

The purpose of a quality plan is to determine a project's quality goals before development starts. This information helps direct the testing and validation effort and determines whether or not a product has achieved its quality goals and, therefore, is ready to be shipped to customers. In the absence of a quality plan, products are often shipped at the schedule's end date without regard to product quality.

A quality plan should state the project quality goals–in measurable terms, when possible. It should define the input and output criteria for each development phase and identify the test and verification activities to be carried out for each phase. The plan should also identify those who are responsible for and have authority for the quality assurance activities performed.

➢ Design and Implementation

The reasons for using a design and implementation methodology include use of common language to express and evaluate designs, consistent approach among projects, and elimination of haphazard development. Projects of different size and complexity may elect to use different design methods and degrees of formality.

However, ISO 9000-3 suggests that a systematic design methodology appropriate for the software product being developed should be identified and used. Design and coding rules and conventions should be identified. Reviews should be carried out to ensure the product requirements are met and the methods identified are employed appropriately.

➢ Testing and Validation

A test plan identifies the levels and types of tests to be run on a product, the resources required, the schedule, and the required inputs and expected outputs for each test case. 9000-3 recognizes that several types of testing may be necessary to adequately exercise a product, such as unit, integration, system, and acceptance testing.

The test plan should include descriptions of test environments, tools, software, and documentation needed. The required version of each software and hardware component in the test environment should be specified. The inputs and expected results for each test case should be documented. Completion criteria for each level of testing should be described.

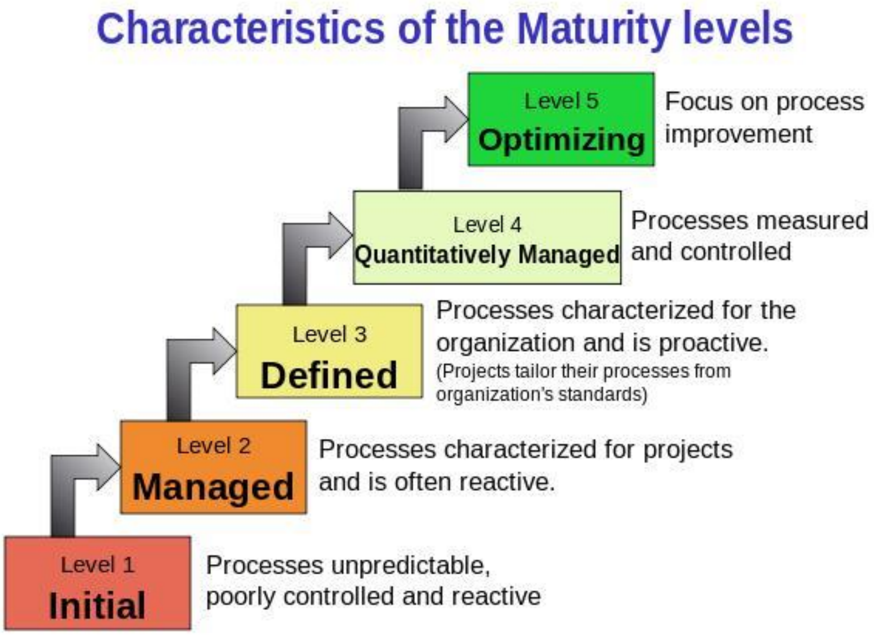
➢ Maintenance

Because the maintenance phase of a product is likely to last much longer than the original product design and development, this phase should consist of a carefully planned set of activities. A maintenance plan should identify support organizations, activities covered under the plan, the records and reports produced, and release procedures for distributing and installing customer updates. The maintenance plan should cover problem resolution, interface modifications, and functional enhancement for the product.

CAPABILITY MATURITY MODEL (CMM)

- Capability Maturity Model is a bench-mark for measuring the maturity of an organization's software process. It is a methodology used to develop and refine an organization's software development process. CMM can be used to assess an organization against a scale of five process maturity levels based on certain Key Process Areas (KPA). It describes the maturity of the company based upon the project the company is dealing with and the clients. Each level ranks the organization according to its standardization of processes in the subject area being assessed.
- A maturity model provides:
    - A place to start
    - The benefit of a community's prior experiences
    - A common language and a shared vision
    - A framework for prioritizing actions
    - A way to define what improvement means for your organization
- In CMMI models with a staged representation, there are five maturity levels designated by the numbers 1 through 5 as shown below:

1. Initial
2. Repeatable / Managed
3. Defined
4. Quantitatively Managed
5. Optimizing


Characteristics of the Maturity levels
Level 5
Optimizing
Focus on process improvement
Level 4
Quantitatively Managed
Processes measured and controlled
Level 3
Defined
Processes characterized for the organization and is proactive.
(Projects tailor their processes from organization's standards)
Level 2
Managed
Processes characterized for projects and is often reactive.
Level 1
Initial
Processes unpredictable, poorly controlled and reactive

- Maturity levels consist of a predefined set of process areas. The maturity levels are measured by the achievement of the **specific** and **generic goals** that apply to each predefined set of process areas. The following sections describe the characteristics of each maturity level in detail.
- Key Process Areas (KPAs) of each level are shown in below table.

CMM Level	Focus	Key Process Ares
1. Initial	Competent people	
2. Repeatable	Project management	Software project planning Software configuration management
3. Defined	Definition of processes	Process definition Training program Peer reviews
4. Managed	Product and process quality	Quantitative process metrics Software quality management
5. Optimizing	Continuous process improvement	Defect prevention Process change management Technology change management

Maturity Level 1 – Initial:

Company has no standard process for software development. Nor does it have a project-tracking system that enables developers to predict costs or finish dates with any accuracy.

- At maturity level 1, processes are usually ad hoc and chaotic.
- The organization usually does not provide a stable environment. Success in these organizations depends on the competence and heroics of the people in the organization and not on the use of proven processes.
- Maturity level 1 organizations often produce products and services that work but company has no standard process for software development. Nor does it have a project-tracking system that enables developers to predict costs or finish dates with any accuracy.
- Maturity level 1 organizations are characterized by a tendency to over commit, abandon processes in the time of crisis, and not be able to repeat their past successes.

Maturity Level 2 – Repeatable / Managed:

Company has installed basic software management processes and controls. But there is no consistency or coordination among different groups.

- At maturity level 2, an organization has achieved all the **specific** and **generic goals** of the maturity level 2 process areas. In other words, the projects of the organization have ensured that requirements are managed and that processes are planned, performed, measured, and controlled.
- The process discipline reflected by maturity level 2 helps to ensure that existing practices are retained during times of stress. When these practices are in place, projects are performed and managed according to their documented plans.
- At maturity level 2, requirements, processes, work products, and services are managed. The status of the work products and the delivery of services are visible to management at defined points.
- Commitments are established among relevant stakeholders and are revised as needed. Work products are reviewed with stakeholders and are controlled.
- The work products and services satisfy their specified requirements, standards, and objectives.

Maturity Level 3 – Defined:

Company has pulled together a standard set of processes and controls for the entire organization so that developers can move between projects more easily and customers can begin to get consistency from different groups.

- At maturity level 3, an organization has achieved all the **specific** and **generic goals.**
- At maturity level 3, processes are well characterized and understood, and are described in standards, procedures, tools, and methods.
- A critical distinction between maturity level 2 and maturity level 3 is the scope of standards, process descriptions, and procedures. At maturity level 2, the standards, process descriptions, and procedures may be quite different in each specific instance of the process (for example, on a particular project). At maturity level 3, the standards, process descriptions, and procedures for a project are tailored from the organization's set of standard processes to suit a particular project or organizational unit.
- The organization's set of standard processes includes the processes addressed at maturity level 2 and maturity level 3. As a result, the processes that are performed across the organization are consistent except for the differences allowed by the tailoring guidelines.
- Another critical distinction is that at maturity level 3, processes are typically described in more detail and more rigorously than at maturity level 2.
- At maturity level 3, processes are managed more proactively using an understanding of the interrelationships of the process activities and detailed measures of the process, its work products, and its services.
- ISO 9000 aims at achieving this level.

Maturity Level 4 – Managed / Quantitatively Managed:

In addition to implementing standard processes, company has installed systems to measure the quality of those processes across all projects.

- At maturity level 4, an organization has achieved all the **specific goals** of the process areas assigned to maturity levels 2, 3, and 4 and the **generic goals** assigned to maturity levels 2 and 3.
- At maturity level 4 Sub-processes are selected that significantly contribute to overall process performance. These selected sub-processes are controlled using statistical and other quantitative techniques.
- Quantitative objectives for quality and process performance are established and used as criteria in managing processes. Quantitative objectives are based on the needs of the customer, end users, organization, and process implementers. Quality and process performance are understood in statistical terms and are managed throughout the life of the processes.
- For these processes, detailed measures of process performance are collected and statistically analysed. Special causes of process variation are identified and, where appropriate, the sources of special causes are corrected to prevent future occurrences.
- Quality and process performance measures are incorporated into the organizations measurement repository to support fact-based decision making in the future.
- A critical distinction between maturity level 3 and maturity level 4 is the predictability of process performance. At maturity level 4, the performance of processes is controlled using statistical and other quantitative techniques, and is quantitatively predictable. At maturity level 3, processes are only qualitatively predictable.

Maturity Level 5 – Optimizing:

Company has accomplished all of the above and can now begin to see patterns in performance over time, so it can tweak its processes in order to improve productivity and reduce defects in software development across the entire organization.

- At maturity level 5, an organization has achieved all the **specific goals** of the process areas assigned to maturity levels 2, 3, 4, and 5 and the **generic goals** assigned to maturity levels 2 and 3.
- Processes are continually improved based on a quantitative understanding of the common causes of variation inherent in processes.
- Maturity level 5 focuses on continually improving process performance through both incremental and innovative technological improvements.
- Quantitative process-improvement objectives for the organization are established, continually revised to reflect changing business objectives, and used as criteria in managing process improvement.
- The effects of deployed process improvements are measured and evaluated against the quantitative process-improvement objectives. Both the defined processes and the organization's set of standard processes are targets of measurable improvement activities.
- Optimizing processes that are agile and innovative depends on the participation of an empowered workforce aligned with the business values and objectives of the organization.
- The organization's ability to rapidly respond to changes and opportunities is enhanced by finding ways to accelerate and share learning. Improvement of the processes is inherently part of everybody's role, resulting in a cycle of continual improvement.
- A critical distinction between maturity level 4 and maturity level 5 is the type of process variation addressed. At maturity level 4, processes are concerned with addressing special causes of process variation and providing statistical predictability of the results. Though processes may produce predictable results, the results may be insufficient to achieve the established objectives. At maturity level 5, processes are concerned with addressing common causes of process variation and changing the process (that is, shifting the mean of the process performance) to improve process performance (while maintaining statistical predictability) to achieve the established quantitative process-improvement objectives.

Shortcomings of CMM

1. Need more guidance to the organisation to improve their quality.
2. The organisation should maintain thicker documents and longer meetings.
3. Getting the accurate measure of an organisation's current maturity level is not feasible.

SUMMARY OF CMM

Maturity Level	**Rating**	**Description**	**KPAs...**
5	Optimizing	Continuous process improvement is enabled by quantitative feedback from the process and from piloting innovative ideas and technologies.	cover the issues that both the organization and the process must address to implement continual, measurable software process improvement. The KPAs are: • Defect Prevention • Technology Change Management • Process Change Management.
4	Managed	Detailed measures of the software process and product quality are collected. Both the software process and products are quantitatively understood and controlled.	focus on establishing a quantitative understanding of both the software process and the software work products being built. The KPAs are: • Quantitative Process Management • Software Quality Management
3	Defined	The software process for both management and engineering activities is documented, standardized and integrated into standard software processes for the organization. All project use an approved, tailored version of the organization's standard software process for developing and maintaining software.	address both project and organizational issues, as the organization establishes an infrastructure that institutionalizes effective software engineering and management processes across all projects. The KPAs are: • Organization Process Focus • Organization Process Definition • Training Program • Integrated Software Management • Software Product Engineering • Intragroup Coordination • Peer Reviews
2	Repeatable	Basic project management processes are established to track cost, schedule, and functionality. The	Focus on the software project's concerns related to establishing

		necessary process discipline is in place to repeat earlier successes on project with similar applications.	basic project management controls. The KPAs are: • Requirements Management • Software Project Planning • Software Project Tracking and Oversight • Software Subcontract Management • Software Quality Assurance • Software Configuration Management
1	Initial	The software process is characterized as ad-hoc, and occasionally even chaotic. Few processes are defined, and success depends on individual effort and heroics.	None

ISO 9000 vs SEI/CMM

The characteristics of ISO 9000 certification and the SEI CMM differ in some respects. The differences are as follows:

- ISO 9000 is awarded by an international standards body. Therefore, ISO 9000 certification can be quoted by an organization in official documents, communication with external parties, and the tender quotations. However, SEI CMM assessment is purely for internal use.
- SEI CMM was developed specifically for software industry and therefore addresses many issues which are specific to software industry alone.
- SEI CMM goes beyond quality assurance and prepares an organization to ultimately achieve Total Quality Management (TQM). In fact, ISO 9001 aims at level 3 of SEI CMM model.
- SEI CMM model provides a list of key process areas (KPAs) on which an organization at any maturity level needs to concentrate to take it from one maturity level to the next. Thus, it provides a way for achieving gradual quality improvement.

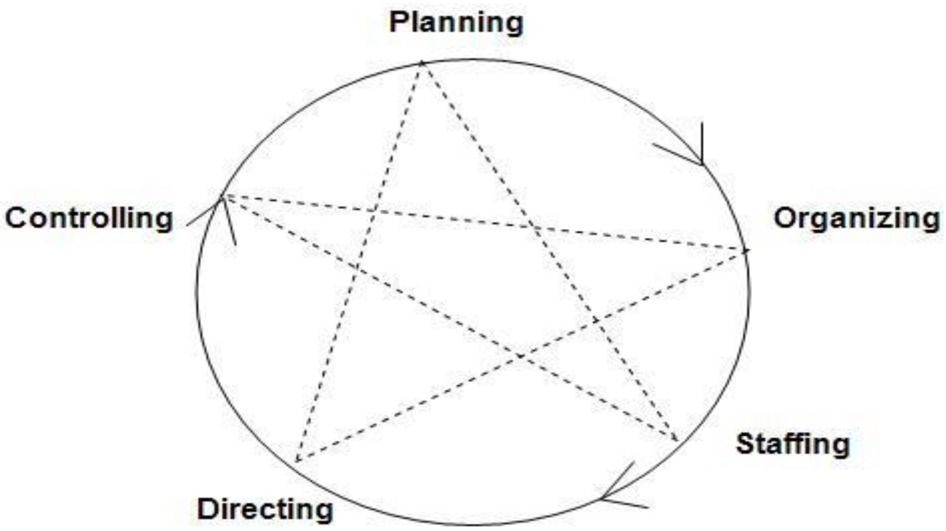
➢ COMPARISON BETWEEN ISO & CMM

ISO	**CMM**
Minimum requirements with implied continuous improvement	Explicit Continuous Quality Improvement
Not specific to any one industry or service	Software specific
Outwardly focused from the firm	Inwardly focused to the firm
Registration Document	No Documentation
Continual Audits	No follow up audits

13. SOFTWARE PROJECT MANAGEMENT

A system of management procedures, practices, technologies, skills, and experience necessary to successfully manage a software project

- The main goal of software project management is to enable a group of software developers to work efficiently towards successful completion of the project.
- The entire SPM consist of five phases: Planning, Staffing, Organizing, Controlling & Directing.



PLANNING

- ✓ It is the basic function of management. It deals with chalking out a future course of action & deciding in advance the most appropriate course of actions for achievement of pre-determined goals.
- ✓ Planning is deciding in advance - what to do, when to do & how to do. It bridges the gap from where we are & where we want to be".
- ✓ A plan is a future course of actions. It is an exercise in problem solving & decision making.
- ✓ Planning is determination of courses of action to achieve desired goals. Thus, planning is a systematic thinking about ways & means for accomplishment of pre-determined goals.

Planning activities

- Set objectives and goals
- Develop strategies
- Develop policies
- Forecast future situations
- Conduct a risk assessment
- Determine possible courses of action
- Make planning decisions
- Set procedures and rules
- Develop project plans
- Prepare budgets
- Document project plans