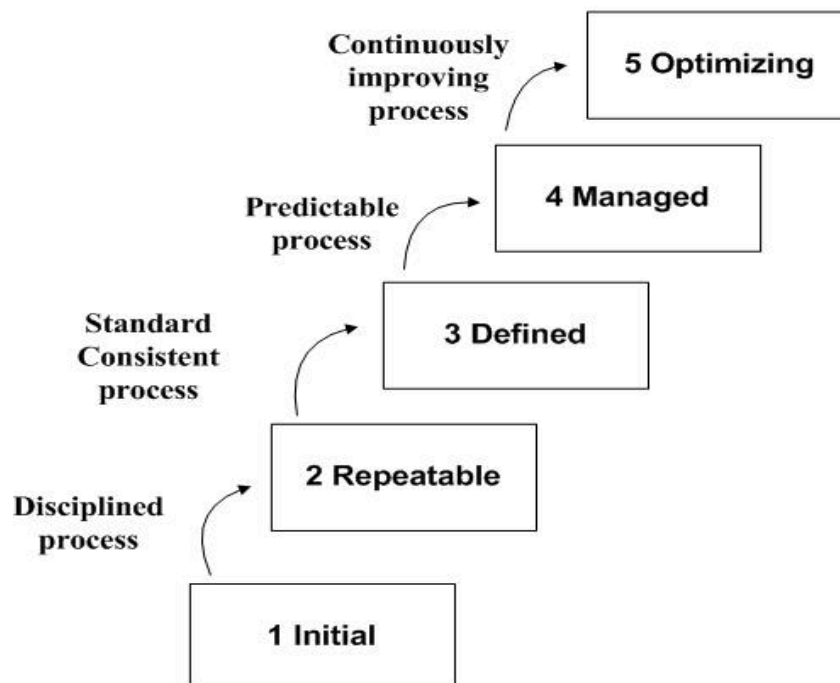## ❖ Capability Maturity Model (CMM) Generic Process Model

Capability Maturity Model (CMM) was developed by the Software Engineering Institute (SEI) at Carnegie Mellon University in 1987. It is not a software process model. It is a framework that is used to analyze the approach and techniques followed by any organization to develop software products. It also provides guidelines to further enhance the maturity of the process used to develop those software products.

It is based on profound feedback and development practices adopted by the most successful organizations worldwide. This model describes a strategy for software process improvement that should be followed by moving through 5 different levels. Each level of maturity shows a process capability level. All the levels except level 1 are further described by Key Process Areas (KPA).

Five Levels of Software Process Maturity in CMM

Continuously improving process → **5 Optimizing**

Predictable process → **4 Managed**

Standard Consistent process → **3 Defined**

Disciplined process → **2 Repeatable**

1 Initial

Principles of Capability Maturity Model (CMM)

➢ People's capability is a competitive issue. Competition arises when different organizations are performing the same task (such as software development). In such a case, the people of

an organization are sources of strategy and skills, which in turn results in better performance of the organization.

➢ The people's capability should be defined in relation to the business objectives of the organization.

➢ An organization should invest in improving the capabilities and skills of the people as they are important for its success.

➢ The management should be responsible for enhancing the capability of the people in the organization.

➢ The improvement in the capability of people should be done as a process. This process should incorporate appropriate practices and procedures.

➢ The organization should be responsible for providing improvement opportunities so that people in the organization can take advantage of them.

➢ Since new technologies and organizational practices emerge rapidly, organizations should continually improve their practices and develop the abilities of people.

Shortcomings of the Capability Maturity Model (CMM)

➢ It encourages the achievement of a higher maturity level in some cases by displacing the true mission, which is improving the process and overall software quality.

➢ It only helps if it is put into place early in the software development process.

➢ It has no formal theoretical basis and in fact, is based on the experience of very knowledgeable people.

➢ It does not have good empirical support and this same empirical support could also be constructed to support other models.

➢ Difficulty in measuring process improvement: The SEI/CMM model may not provide an accurate measure of process improvement, as it relies on self-assessment by the organization and may not capture all aspects of the development process.

➢ Focus on documentation rather than outcomes: The SEI/CMM model may focus too much on documentation and adherence to procedures, rather than on actual outcomes such as software quality and customer satisfaction.

PARSHWANATH CHARITABLE TRUST'S
A.P. SHAH INSTITUTE OF TECHNOLOGY
Department of Computer Science and Engineering
Data Science

CSE DATA SCIENCE

➢ May not be suitable for all types of organizations: The SEI/CMM model may not be suitable for all kinds of organizations, particularly those with smaller development teams or those with less structured development processes.

➢ May not keep up with rapidly evolving technologies: The SEI/CMM model may not be able to keep up with rapidly evolving technologies and development methodologies, which could limit its usefulness in certain contexts.

➢ Lack of agility: The SEI/CMM model may not be agile enough to respond quickly to changing business needs or customer requirements, which could limit its usefulness in dynamic and rapidly changing environments.

Key Process Areas (KPA)

Each of these KPA (Key Process Areas) defines the basic requirements that should be met by a software process in order to satisfy the KPA and achieve that level of maturity.

Conceptually, key process areas form the basis for management control of the software project and establish a context in which technical methods are applied, work products like models, documents, data, reports, etc. are produced, milestones are established, quality is ensured and change is properly managed.

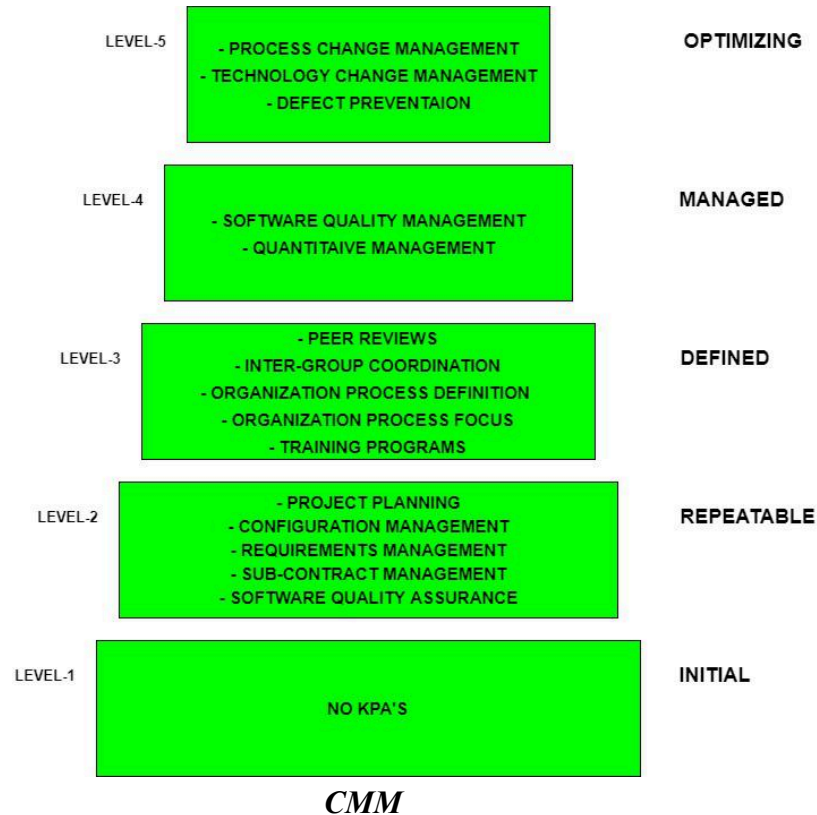**Levels of Capability Maturity Model (CMM)**

There are basically 5 levels of Capability Maturity Models.

LEVEL-5	- PROCESS CHANGE MANAGEMENT - TECHNOLOGY CHANGE MANAGEMENT - DEFECT PREVENTAION	OPTIMIZING
LEVEL-4	- SOFTWARE QUALITY MANAGEMENT - QUANTITAIVE MANAGEMENT	MANAGED
LEVEL-3	- PEER REVIEWS - INTER-GROUP COORDINATION - ORGANIZATION PROCESS DEFINITION - ORGANIZATION PROCESS FOCUS - TRAINING PROGRAMS	DEFINED
LEVEL-2	- PROJECT PLANNING - CONFIGURATION MANAGEMENT - REQUIREMENTS MANAGEMENT - SUB-CONTRACT MANAGEMENT - SOFTWARE QUALITY ASSURANCE	REPEATABLE
LEVEL-1	NO KPA'S	INITIAL

CMM

Level-1: Initial

● No KPIs defined.

● Processes followed are Ad Hoc and immature and are not well defined.

● Unstable environment for software development.

● No basis for predicting product quality, time for completion, etc.

● Limited project management capabilities, such as no systematic tracking of schedules, budgets, or progress.

● We have limited communication and coordination among team members and stakeholders.

- No formal training or orientation for new team members.

- Little or no use of software development tools or automation.

- Highly dependent on individual skills and knowledge rather than standardized processes.

- High risk of project failure or delays due to a lack of process control and stability.

Level-2: Repeatable

- Focuses on establishing basic project management policies.

- Experience with earlier projects is used for managing new similar-natured projects.

- Project Planning- It includes defining resources required, goals, constraints, etc. for the project. It presents a detailed plan to be followed systematically for the successful completion of good-quality software.

- Configuration Management- The focus is on maintaining the performance of the software product, including all its components, for the entire lifecycle.

- Requirements Management- It includes the management of customer reviews and feedback which result in some changes in the requirement set. It also consists of accommodation of those modified requirements.

- Subcontract Management- It focuses on the effective management of qualified software contractors i.e. it manages the parts of the software developed by third parties.

- Software Quality Assurance- It guarantees a good quality software product by following certain rules and quality standard guidelines while developing.

![A.P. Shah Institute of Technology header]

PARSHWANATH CHARITABLE TRUST'S

A.P. SHAH INSTITUTE OF TECHNOLOGY

Department of Computer Science and Engineering

Data Science

CSE DATA SCIENCE

Level-3: Defined

● At this level, documentation of the standard guidelines and procedures takes place.

● It is a well-defined integrated set of project-specific software engineering and management processes.

● Peer Reviews: In this method, defects are removed by using a number of review methods like walkthroughs, inspections, buddy checks, etc.

● Intergroup Coordination: It consists of planned interactions between different development teams to ensure efficient and proper fulfillment of customer needs.

● Organization Process Definition: Its key focus is on the development and maintenance of standard development processes.

● Organization Process Focus: It includes activities and practices that should be followed to improve the process capabilities of an organization.

● Training Programs: It focuses on the enhancement of knowledge and skills of the team members including the developers and ensuring an increase in work efficiency.

Level-4: Managed

● At this stage, quantitative quality goals are set for the organization for software products as well as software processes.

● The measurements made help the organization to predict the product and process quality within some limits defined quantitatively.

● Software Quality Management: It includes the establishment of plans and strategies to develop quantitative analysis and understanding of the product's quality.

- Quantitative Management: It focuses on controlling the project performance in a quantitative manner.

Level-5: Optimizing

- This is the highest level of process maturity in CMM and focuses on continuous process improvement in the organization using quantitative feedback.

- The use of new tools, techniques, and evaluation of software processes is done to prevent the recurrence of known defects.

- Process Change Management: Its focus is on the continuous improvement of the organization's software processes to improve productivity, quality, and cycle time for the software product.

- Technology Change Management: It consists of the identification and use of new technologies to improve product quality and decrease product development time.

- Defect Prevention It focuses on the identification of causes of defects and prevents them from recurring in future projects by improving project-defined processes.