PROCESS MODEL

A PROCESS MODEL IS A STRUCTURED COLLECTION OF PRACTICES THAT DESCRIBE THE CHARACTERISTICS OF EFFECTIVE PROCESSES, PROVEN BY EXPERIENCE.

A PROCESS MODAL IS USED:

- TO SET MEASURABLE GOALS AND PRIORITIES.
- TO ENSURE STABLE, CAPABLE AND MATURE PROCESSES.
- AS A GUIDE FOR IMPROVEMENT OF PROJECTS AND ORGANIZATIONAL PROCESSES.
- TO DIAGNOSE AND CERTIFY EXISTING ORGANIZATIONAL PRACTICES.

CAPABILITY MATURITY MODEL INTEGRATION

CMMI IS A PROCESS IMPROVEMENT APPROACH THAT PROVIDES ORGANIZATIONS WITH THE ESSENTIAL ELEMENTS OF EFFECTIVE PROCESSES.

CMMI CAN BE USED IN PROCESS IMPROVEMENTS AS A:

- COLLECTIONS OF BEST PRACTICES.
- FRAMEWORK FOR ORGANIZING AND PRIORITIZING ACTIVITIES.

THE CAPABILTY LEVELS IN THE CMMI FRAMEWORK ARE USED TO MEASURE THE MATURITY AND QUALITY OF A SPECIFIC PROCESS WITHIN AN ORGANIZATION. THEY ALLOW FOR THE ASSESSMENT OF HOW STABLE, EFFECTIVE, AND PREDICTABLE A PROCESS IS, HELPING COMPANIES IMPROVE THEIR WORKING METHODS:

LEVEL 0 - INCOMPLETE:
- THE PROCESS IS EITHER NOT PERFORMED OR ONLY PARTIALLY PERFORMED.
- SPECIFIC GOALS OF THE PROCESS ARE NOT SATISFIED.

LEVEL 1 - PERFORMED:
- THE PROCESS SATISFIES THE SPECIFIC GOALS.

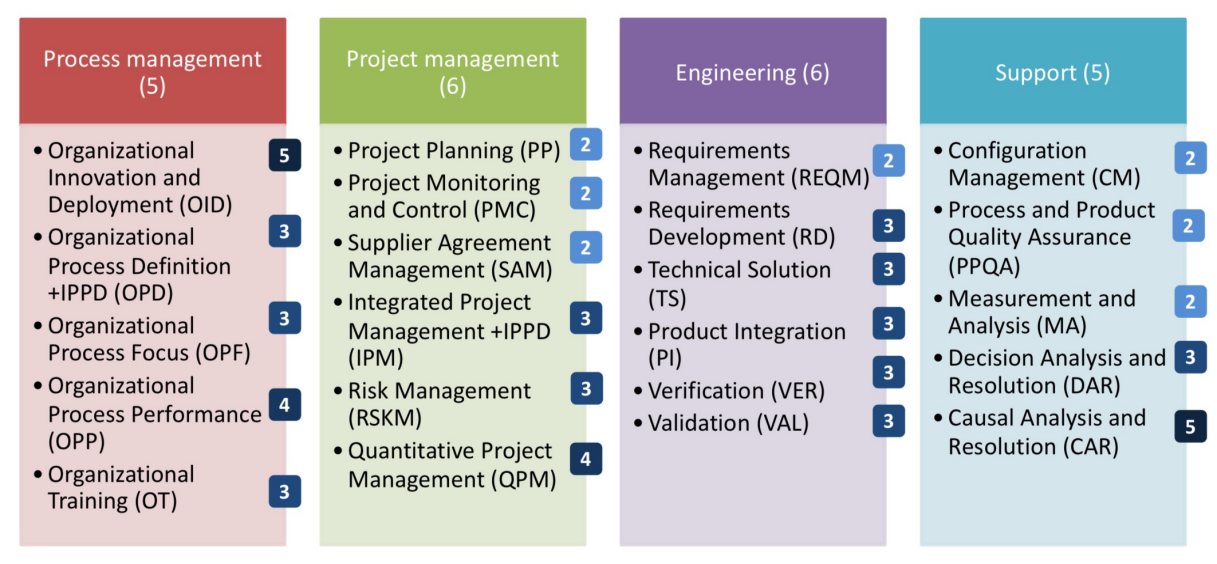
LEVEL 2 - MANAGED:
- THE PROCESS IS PLANNED, MONITORED, AND CONTROLLED.
- NECESSARY RESOURCES AND SKILLS ARE ALLOCATED

LEVEL 3 - DEFINED:
- THE PROCESS IS STANDARDIZED AND TAILORED TO THE ORGANIZATION'S NEEDS.
- IT CONTRIBUTES TO THE OVERALL IMPROVEMENT OF ORGANIZATIONAL PROCESSES.

LEVEL 4 - QUANTITATIVELY MANAGED:
- THE PROCESS IS CONTROLLED USING STATISTICAL AND QUANTITATIVE TECHNIQUES.
- QUALITY OBJECTIVES ARE MEASURED AND USED TO MANAGE THE PROCESS.

LEVEL 5 - OPTIMIZING:
- THE PROCESS IS CONTINUOSLY IMPROVED THROUGH INNOVATIONS
- THE FOWS IS ON PREVENTING ISSUES BEFORE THEY ARISE.

ALL CMMI MODELS CONTAIN MULTIPLE PROCESS AREAS (PAs) THERE ARE 22 AREAS DIVIDED INTO PROCESS MANAGEMENT, PROJECT MANAGEMENT, ENGINEERING AND SUPPORT.



ISO 12207

IT DEFINES AND STRUCTURES ALL ACTIVITIES INVOLVED IN THE SOFTWARE DEVELOPMENT LIFECYCLE. ITS MAIN GOAL IS TO PROVIDE A COMMON LANGUAGE TO INVOLVED STAKEHOLDERS.
THE STANDARD ADOPTS A FUNCTIONAL APPROACH, ORGANIZING ACTIVITIES AS A SET OF COORDINATED ACTIONS THAT TRASFORM INPUTS INTO OUTPUTS.

THE PRIMARY LIFECYCLE PROCESSES ARE: ACQUISITION, DEVELOPMENT, OPERATION AND MAINTENANCE.

THE SUPPORT PROCESSES ARE: CONFIGURATION MANAGEMENT, QUALITY ASSURANCE, VALIDATION, AUDITS AND PROBLEM SOLVING.

THE ORGANIZATIONAL PROCESSES ARE: MANAGEMENT, INFRASTRUCTURE, IMPROVEMENT AND TRAINING.

THE STANDARD IS BASED ON TWO BASIC PRINCIPLES.

MODULARITY. PROCESSES WITH MINIMUM COUPLING AND MAXIMUM COHESION.

RESPONSIBILITY: ASSIGNING CLEAR RESPONSIBILITIES FOR EACH PROCESS TO FACILITATE APPLICATION IN COMPLEX PROJECTS INVOLVING MULTIPLE STAKEHOLDERS.

ISO 9000

GOALS:

- IMPROVE EFFICIENCY.
- ENSURE PRODUCTS AND SERVICES MEET CUSTOMER EXPECTATIONS
- CONTINUOUS IMPROVEMENT OF QUALITY MANAGEMENT SYSTEMS.

THE MAIN COMPONENTS ARE:

QUALITY MANAGEMENT SYSTEM: GENERAL REQUIREMENTS AND DOCUMENTATION

MANAGEMENT RESPONSIBILITY: INTERNAL PLANNING AND COMMUNICATION.

RESOURCE MANAGEMENT: ADEQUATE PEOPLE AND PHYSICAL RESOURCES.

PRODUCT/SERVICE REALIZATION: CONVERTING INPUTS TO OUTPUTS.

MEASUREMENT AND IMPROVEMENT: AUDIT AND ANALYSIS TO IMPROVE SYSTEMS.

ISO CERTIFICATION

ISO DOESN'T ITSELF CERTIFY ORGANIZATIONS BUT WORKS THROUGH ACCREDITED BODIES THAT AUTHORIZE CERTIFICATION AGENCIES.

A CERTIFICATION IS NOT PERMANENT; IT MUST BE RENEWED PERIODICALLY.

QUALITY REQUIREMENTS

THE QUALITY MANUAL (QM) DEFINES AN ORGANIZATION'S QUALITY POLICIES AND PROCESSES.

THE QUALITY POLICY (QP) IS A SPECIFIC ADAPTATION OF THE QM FOR PROJECTS.

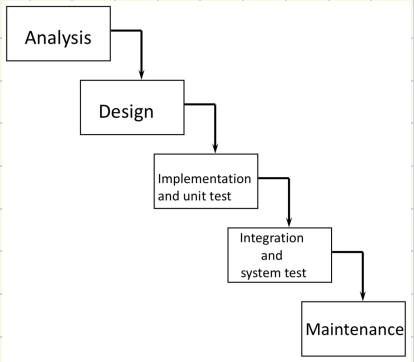
SOFTWARE PROCESS MODELS

THE SOFTWARE DEVELOPMENT PROCESS IS THE HEART OF SOFTWARE ENG. DUE TO THE INTANGIBILITY OF SW PRODUCTS, SPECIFIC METHODS MUST BE ADOPTED TO MANAGE PROJECTS.
PROCESS MODELS SERVE TO CLEARLY DEFINE THE ACTIVITIES TO BE CARRIED OUT AND THE DOCUMENTATION TO BE PRODUCED.

GENERIC SW PROCESS MODELS

GENERIC PROCESS MODELS DEFINE DIFFERENT STRATEGIES FOR APPROACHING SW DEVELOPMENT:

THE WATERFALL MODEL: DIVIDES THE WORK INTO SEQUENTIAL PHASES. ALTHOUGH IT IS STRUCTURED, IT SUFFERS FROM RIGIDITY AND DIFFICULTY IN ADAPTING TO CHANGES.



EVOLUTIONARY DEVELOPMENT: SPECIFICATION, DEVELOPMENT AND VALIDATION ARE INTERLEAVED.

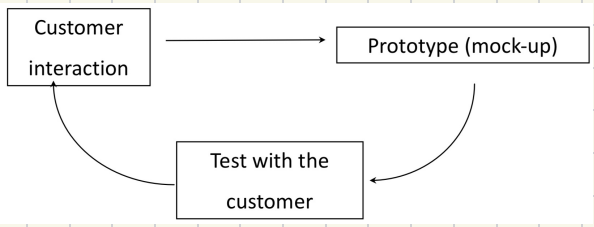
COMPONENT-BASED SW ENG: LEVERAGE PRE-EXISTING COMPONENTS TO ACCELERATE DEVELOPMENT.

FORMAL MODELS: REQUIREMENTS ARE EXPRESSED IN A FORMAL LANGUAGE.

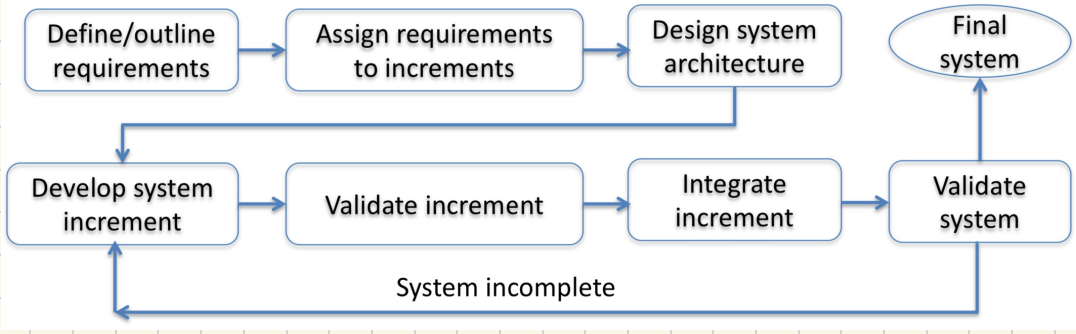
PROCESS ITERATION

SOFTWARE SYSTEM REQUIREMENTS CONTINUALLY EVOLVE, REQUIRING ITERATIONS IN THE DEVELOPMENT PROCESS
THESE ITERATIVE APPROACHES INCLUDE:

PROTOTYPAL MODEL: PRELIMINARY VERSIONS DEVELOPED TO CLARIFY REQUIREMENTS THROUGH CUSTOMER INTERACTION.



INCREMENTAL DEVELOPMENT: DELIVERY OF WORKING INTERMEDIATE VERSIONS

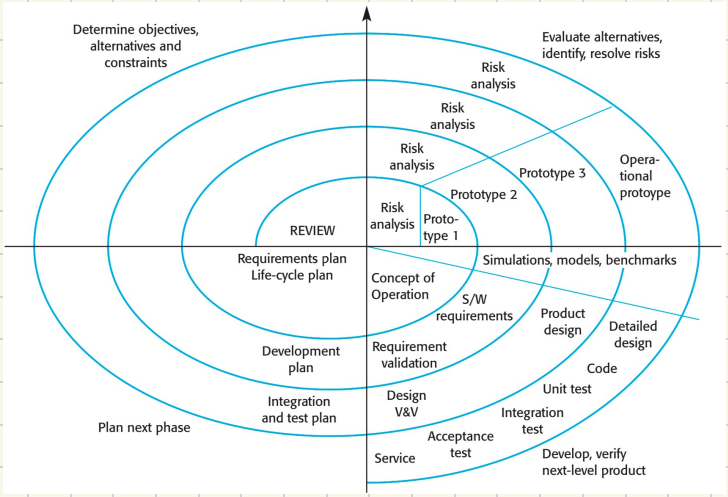


EXTREME PROGRAMMING

XP EMPHASIZES SMALL INCREMENTS OF FUNCTIONALITY, DIRECTLY INVOLVING THE CUSTOMER AND CONTINUOSLY IMPROVING THE CODE, WITH A COLLABORATIVE AND ITERATIVE APPROACH.

SPIRAL DEVELOPMENT

THE SPIRAL MODEL INTEGRATES FLEXIBLE PHASES AND ACTIVE RISK MANAGEMENT.
EACH LOOP IN THE SPIRAL REPRESENTS A GOAL, RISK ASSESSMENT, DEVELOPMENT AND SUBSEQUENT PLANNING.



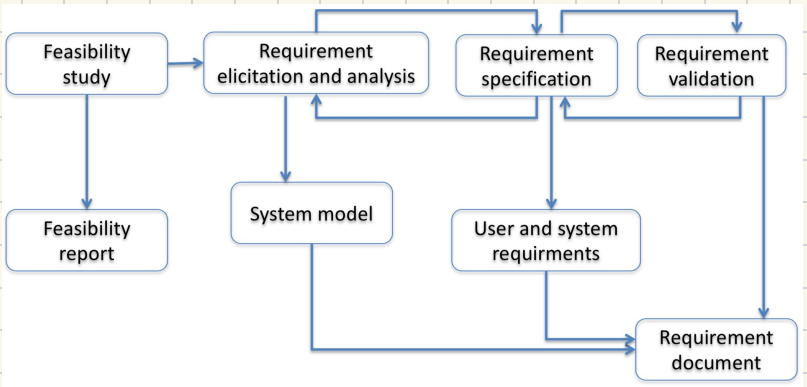
CORE PROCESS ACTIVITIES

THE MAIN INVOLVED ACTIVITY ARE: SW SPEC, SW DESIGN AND IMPL, PROG AND DEBUG, SW VAL, AND TEST.

SOFTWARE SPECIFICATION

THE PROCESS OF ESTABLISHING WHAT SERVICES ARE REQUIRED AND THE CONSTRAINTS ON THE SYSTEM OPERATION AND DEVELOPMENT.
INCLUDES STEPS SUCH AS:

- FEASIBILITY STUDY
- ANALYSIS AND SPECIFICATION OF REQUIREMENTS.
- VALIDATION OF REQUIREMENTS



SOFTWARE DESIGN AND IMPLEMENTATION

THE PROCESS OF CONVERTING THE SYSTEM SPECIFICATION INTO A EXECUTABLE SYSTEM.
ACTIVITIES INCLUDE:

- ARCHITECTURAL DESIGN, INTERFACES, COMPONENTS AND ALGORITHMS.
- USE OF GRAPHICAL MODELS, LIKE UML, TO DOCUMENT THE DESIGN.

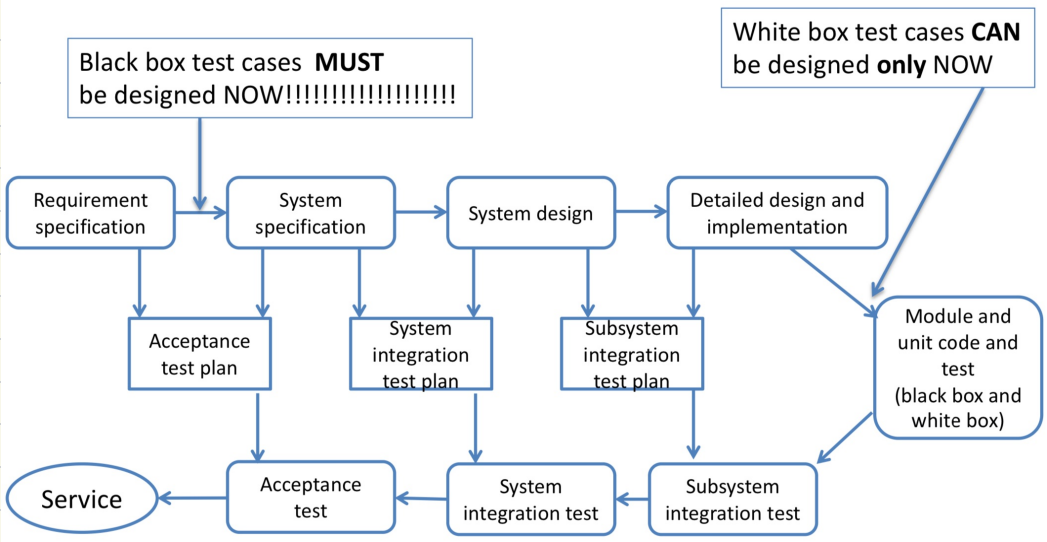
PROGRAMMING AND DEBUGGING

TURN DESIGN INTO CODE AND REMOVE ERRORS THROUGH TESTING AND FIXING.

SOFTWARE VALIDATION

VERIFICATION AND VALIDATION TO ENSURE COMPLIANCE WITH SPECIFICATIONS AND WSTOMER SATISFACTIONS.
TESTS INCLUDE.

- UNIT TESTING: TESTS ON INDIVIDUAL COMPONENTS.
- SYSTEM TESTING: OVERALL INTEGRATION AND TESTING.
- ACCEPTANCE TESTING: TESTING WITH REAL DATA TO CONFIRM REQUIREMENTS.



SOFTWARE EVOLUTION

SW IS FLEXIBLE AND MUST ADAPT TO CHANGING REQUIREMENTS, MAKING ONGOING MAINTENANCE CRITICAL.