GALGOTIAS UNIVERSITY

School of Computing Science and Engineering

Program: Btech

Course Code: BCSE2351

Course Name: Software Project Management

Dr. Bharat Bhushan Naib

Course Prerequisites

- Basics of Software Engineering,
- Software Testing, Software Quality and Management

SYLLABUS

Unit I: Introduction and Software Project Planning
Fundamentals of Software Project Management (SPM), Need Identification, Vision and Scope document, Project Management Cycle, SPM Objectives, Management Spectrum, SPM Framework, Software Project Planning, Planning Objectives, Project Plan, Types of project plan, Structure of a Software Project Management Plan, Software project estimation, Estimation methods, Estimation models, Decision process.
Module II: Project Organization and Scheduling
Project Elements, Work Breakdown Structure (WBS), Types of WBS, Functions, Activities and Tasks, Project Life Cycle and Product Life Cycle, Ways to Organize Personnel, Project schedule, Scheduling Objectives, Building the project schedule, Scheduling terminology and techniques, Network Diagrams: PERT, CPM, Bar Charts: Milestone Charts, Gantt Charts.
Module III: Project Monitoring and Control
Dimensions of Project Monitoring & Control, Earned Value Analysis, Earned Value Indicators: 23 Budgeted Cost for Work Scheduled (BCWS), Cost Variance (CV), Schedule Variance (SV),Cost Performance Index (CPI), Schedule Performance Index (SPI), Interpretation of Earned Value Indicators, Error Tracking, Software Reviews, Types of Review: Inspections, Deskchecks, Walk through, Code Reviews, Pair Programming.
Module IV: Software Quality Assurance and Testing
Testing Objectives, Testing Principles, Test Plans, Test Cases, Types of Testing, Levels of Testing, Test Strategies, Program Correctness, Program Verification & validation, Testing Automation & Testing Tools, Concept of Software Quality, Software Quality Attributes, Software Quality Metrics and Indicators, The SEI Capability Maturity Model CMM), SQA Activities, Formal SQA Approaches: Proof of correctness, Statistical quality assurance, Clean room process.
Module V: Project Management and Project Management Tools
Software Configuration Management: Software Configuration Items and tasks, Baselines, Plan for Change, Change Control, Change Requests Management, Version Control, Risk Management: Risks and risk types, Risk Breakdown Structure (RBS), Risk Management Process: Risk identification, Risk analysis, Risk planning, Risk monitoring, Cost Benefit Analysis, Software Project Management Tools: CASE Tools, Planning and Scheduling Tools, MS-Project.
Module VI: Trends and Application of SPM
Case study - Agile project management with Scrum project- Resource interdependence and project termination: An analysis in the biopharmaceutical industry, Agile Approach, The expansion of Artificial Intelligence (AI) & Automation, Hybrid project management approaches, Advanced project management tools and solutions , Constantly Changing Digital Technologies, Change Management to Project Success, Risk Management has Higher Stakes, Shifting, Globalized Gig Economy

Recommended Books

Text Book
Clifford F. Gray, Erik W. Larson, "Project Management: The Managerial Process with MS", Mc Graw Hill

Reference Book (s):
1. M. Cotterell, Software Project Management, Tata McGraw-Hill Publication.
2. Royce, Software Project Management, Pearson Education
3. Kieron Conway, Software Project Management, Dreamtech Press
4. S. A. Kelkar, Software Project Management, PHI Publication.
5.Bob Hughes and Mike Cotterell, "Software Project Management ", Tata McGraw Hill, 4th edition, 2006
6.Roger S. Pressman , "Software Engineering – A Practitioner's approach", Tata McGraw Hill, 2009

What is a Project?

A project is a temporary endeavor undertaken to create a unique product or service

- A Guide to the Project Management Body of Knowledge (PMBOK™), Project Management Institute, 2003

- **One time (non-routine)**
- **Limited funds/time (May be single phase or multiple phases)**
- **Specific resources utilized (may involve several specialism)**
- **Performed by people - Single or multi-person & multi location team(s)**
- **Planned, controlled (planning, monitoring and controlling is required)**
- **Specific Deliverables (specific objectives are to be met or a specified product is to be created)**

Characteristics of projects

- Non-routine tasks are involved
- Planning is required
- Specific objectives are to be met or a specific product is to be created
- The project has a predetermined time span
- Work is carried out for someone other than yourself
- Work involves several specialisms
- People are formed into a temporary work group to carry out the task
- Work is carried out in several phases
- The resources that are available for use on the project are constrained
- The project is large or complex

What is Management

- *Planning* – deciding what is to be done
- *Organizing* – making arrangements
- *Staffing* – selecting right people for the job etc.
- *Directing* - giving instructions
- *Monitoring* – checking on progress
- *Controlling* – taking action to remedy hold-ups
- *Innovating* – coming up with new solutions
- *Representing* – liaising with clients, users, developers, suppliers and other stakeholders.

The Management Spectrum

Effective project management focuses on four P's (in the order):

The People: Stakeholders, the team leaders, and the software team

- Deals with the cultivation of motivated, highly skilled people and teams
- Includes recruiting, selection, performance management, training, compensation, career development, organization and work design, and team culture development

The Problem/ Product: before a project can be planned

- Its objectives and scope should be established;
- Alternative solutions should be considered; and
- Technical and management constraints should be identified.

The process: a software process provides the framework from which a comprehensive plan for software development can be established.

The project: Planning and controlling a software project is done for one primary reason…it is the only known way to manage complexity

Project Management Skills

- Leadership
- Communications
- Problem Solving
- Negotiating
- Influencing the Organization
- Mentoring
- Process and technical expertise

INTRODUCTION/ FUNDAMENTALS

Software Project Management

- *Software project management* is aimed to ensure that the software is delivered on time, within budget and schedule constraints, and satisfies the requirements of the client

- Management of software projects is different from other types of management because:
  - Software is not tangible
  - Software processes are relatively new and still "under trial"
  - Larger software projects are usually "one-off" projects
  - Computer technology evolves very rapidly

INTRODUCTION/ FUNDAMENTALS

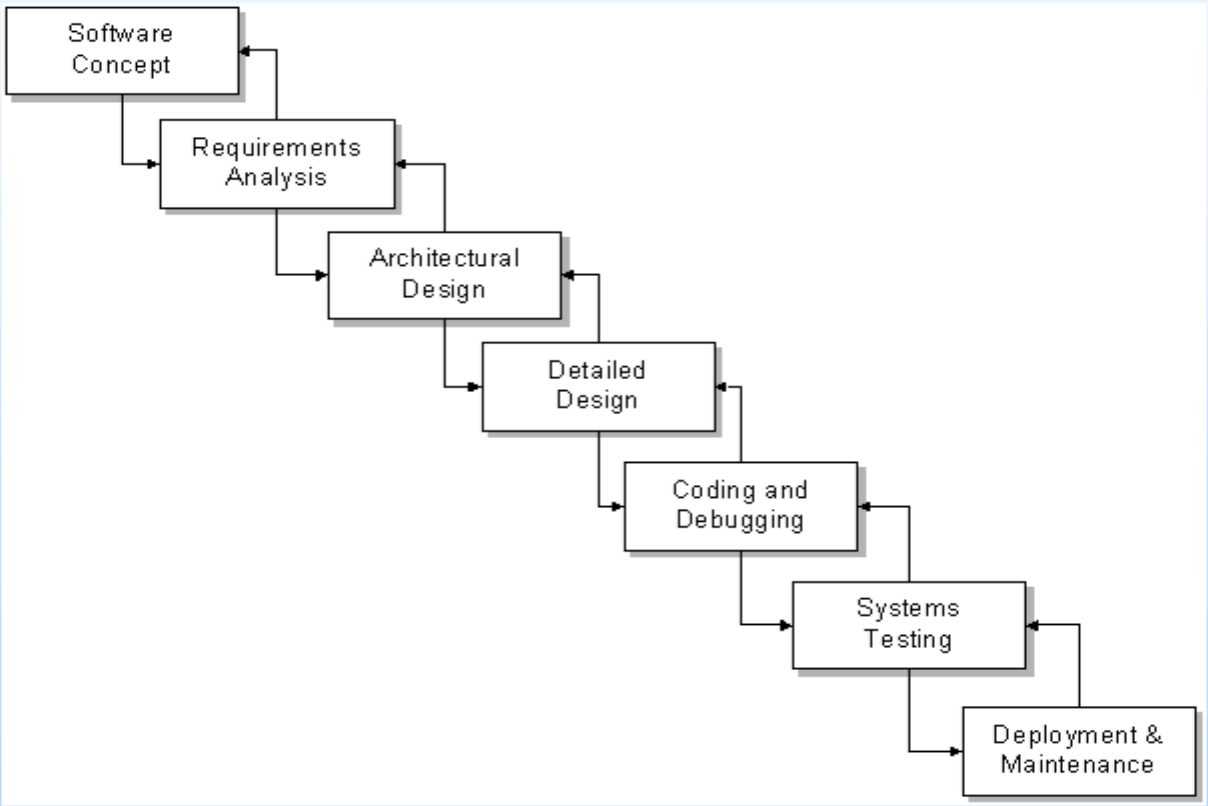
Software Projects versus other types of Projects

- *Invisibility* – With physical artifacts, measuring progress is easy as it can be seen/ felt. However with Software, progress is not immediately visible.
- *Complexity* – Software products are, generally, more complex than other engineering artifact of same value.
- *Flexibility* - It is easier to change/ modify software systems to meet changing organizational/ product requirement as compared to other engineering artifacts; it may not be possible to modify a physical artifact at all.

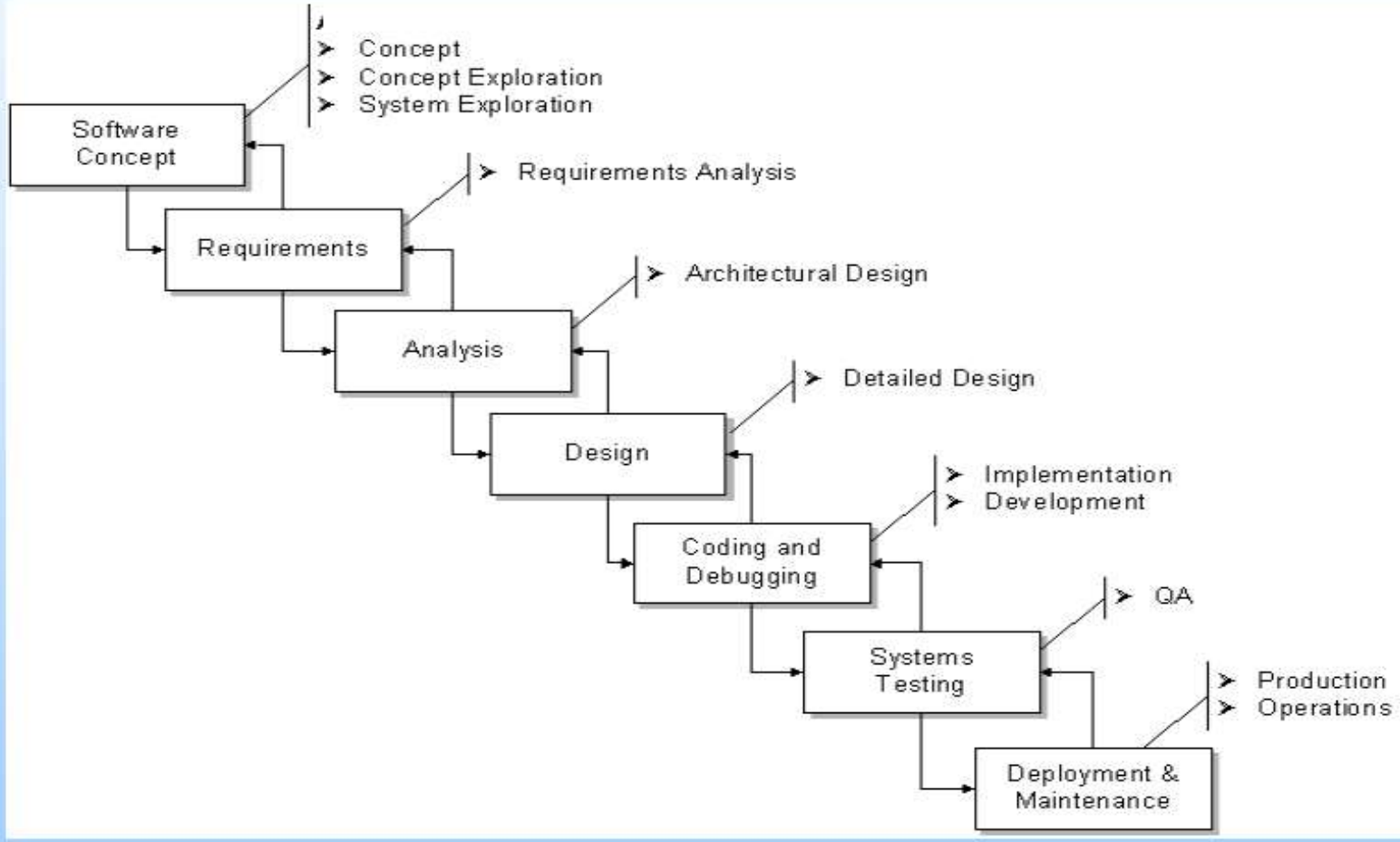
Project Phases

- All projects are divided into phases
- All phases together are known as the Project Life Cycle
- Each phase is marked by completion of Deliverables
- Identify the primary software project phases

Seven Core Project Phases



Project Phases



Project Life Cycle

- **Initiation** – on the first stage, the necessity, feasibility, scope, time, budget and critical success factors of the project are defined along with the approach and methods to be used to deliver the required products and results.

- **Planning** – this stage includes a detailed identification of all the project elements and matters including project team, specified allocation of project resources and timeline, assignments of project tasks, evaluation of risks, definition of criteria for quality and successful completion of each deliverable, etc.

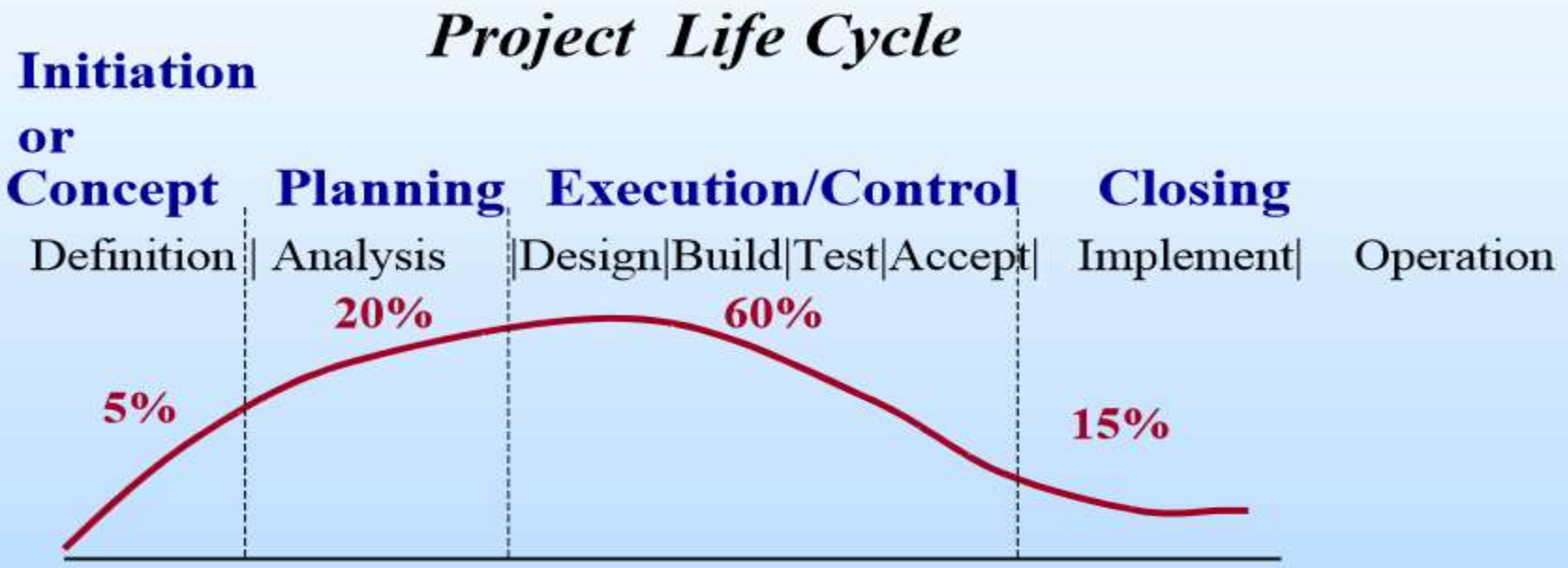
- **Execution** – this is the working phase where the project plan is implemented through practical actions that lead to successful project accomplishment. It is necessary to control performance and quality of all the required activities to know if they match the project's requirements.

- **Closure** – this stage identifies the project completion including testing, evaluation and formal acceptance of the final product by the customer,

Purpose of Project Management

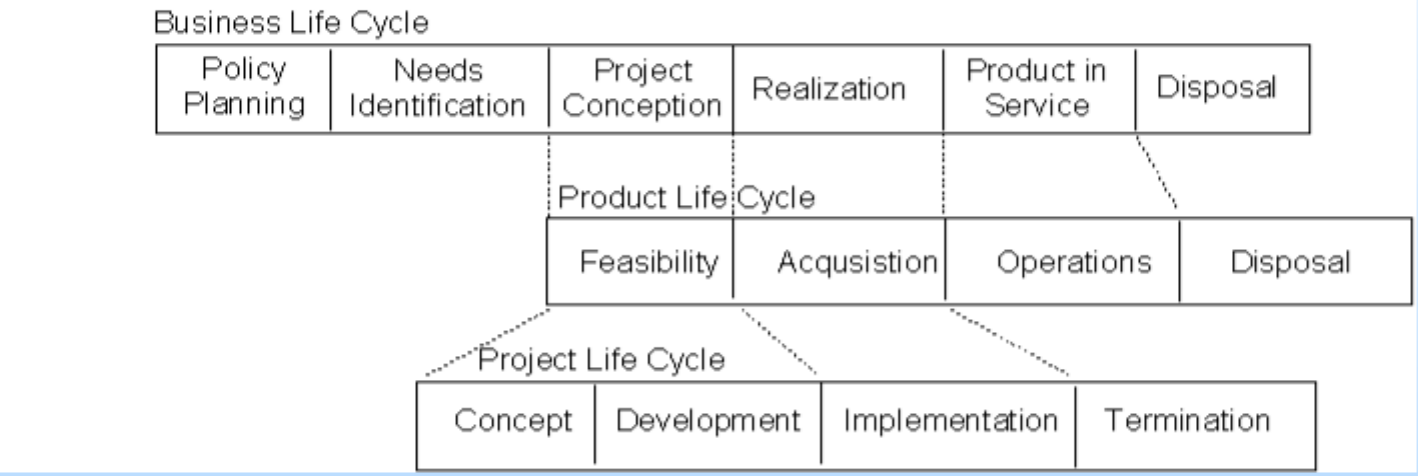
- Ensure meeting the project objectives within the allocated schedule & budget
  - Communication
  - Meetings
  - Reviews
  - Authorization
  - Record Keeping
  - Monitoring (testing)
  - Interface Control

INTRODUCTION/ FUNDAMENTALS



Percentages and graph refer to the amount of *effort (people)*
In IT projects = 90-95% of cost!

Lifecycle Relationships



Project vs. Program Management

- What's a 'program'?
- Mostly differences of scale
- Often a number of related projects
- Longer than projects
- Definitions vary
- Ex: Program Manager for MS Word

INTRODUCTION/ FUNDAMENTALS

The Triple Constraint of Projects

- **On Time, Budget, Quality = Required Scope**

- **Integration**
- **Trade – Off's**

Problems with Software Projects

Manager's Point of View:

- Poor estimates and plans
- Lack of quality standards and measures
- Lack of guidance about making organizational decisions
- Lack of techniques to make progress visible
- Poor role definition – who does what?
- Incorrect success criteria

Members' Point of View:

- Inadequate specification of work
- Management ignorance of IT
- Lack of knowledge of application area
- Lack of standards

Problems with Software Projects

- Lack of standards
- Lack of up-to-date documentation
- Preceding activities not completed on time – including late delivery of equipment
- Lack of communication between users and technicians
- Lack of communication leading to duplication of work
- Lack of commitment – especially when a project is tied to one person who then moves
- Narrow scope of technical expertise
- Changing statutory requirements
- Changing software environment
- Deadline pressure
- Lack of quality control
- Remote management
- Lack of training

Lifecycle Relationships