Software Project Management (SPM)

▶

Course Code: CACS407
Year/ Semester: IV/VII

Compiled by Shishir Ghimire

Credit Hours: 3hrs

▶ **Unit - 04: Software Evaluation & Costing**

Class Load : 8 hrs

BCA 7th Sem - Shishir Ghimire

▶ TABLE OF CONTENTS

Unit -4   **8 Hrs**

Software Evaluation and Costing

Project Evaluation: Strategic Assessment, Technical Assessment, cost-benefit analysis, Cash flow forecasting, cost-benefit evaluation techniques, Risk Evaluation. Selection of Appropriate Report, Project approach: Choosing technologies, choice of process models, structured methods.

▶ Project Portfolio Management:

❖ **Resource Management:**

➢ Organizations have limited resources and should select **projects that align with their goals and mission** for maximum benefit to the organization and society.

❖ **Overview of Projects:**

➢ PPM provides an **overview of all ongoing or potential projects** in the organization.

❖ **Why do we need Project Portfolio Management?**

➢ **Resource Allocation and Project Evaluation:**

■ It helps prioritize resource allocation to projects and decide which new projects to **accept and which existing ones to discontinue** based on evaluation criteria.

▶ Project Portfolio Management:

The concerns of project portfolio management include:

- ❖ **Identifying** which project proposals are worth implementation;

- ❖ **Assessing** the amount of risk of failure that a potential project has;

- ❖ **Deciding** how to share limited resources, including staff time and finance, between projects – one problem can be that too many projects are started given the resources available so that inevitably some projects will miss planned completion dates;

- ❖ **Being aware of the dependencies** between projects, especially where several projects need to be completed for an organization to reap benefits;

- ❖ Ensuring that projects **do not duplicate work**;

- ❖ Ensuring that necessary developments **have not been** inadvertently missed.

Project Evaluation

4.1

▶ Project Evaluation (Project Analysis):

❖ **What?**

➢ Whether a project is **worth** undertaking or not

➢ What **factors** are important in **selecting or rejecting** a project proposal before the project is started

❖ **What is Project Evaluation?**

➢ A systematic and objective assessment of project 's performance throughout its lifecycle, enabling you to understand its effectiveness, identify areas of improvement and inform future decisions.
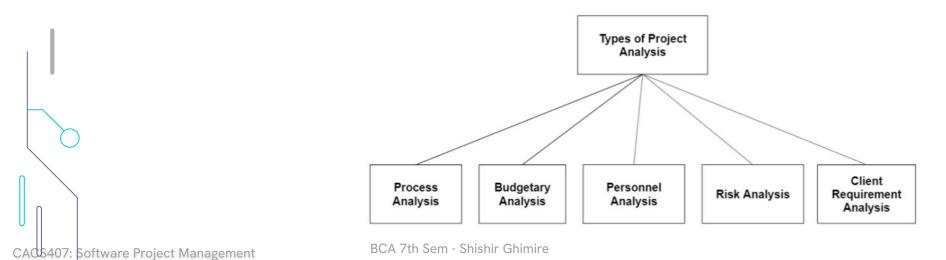
❖ **Why Project Evaluation?**

➢ Want to decide whether a project can proceed before it is too late → Want to decide which of the several alternative projects has a better success rate, a higher turnover, a higher rate of return

▶ Project Evaluation (Project Analysis):

For Example:

Given three projects A, B, C, not all projects can be taken by a particular organization, need to choose **one among three**, how and by what criteria?

❖ **Types of Project Analysis:**

▶ **Project Evaluation** > **Types:**

❖ **Process Analysis:**

➢ Process Analysis keeps checking **whether the tasks are being completed within the allocated time or not**. If they are being delayed we will have to find out the reasons as well.

❖ **Budget Analysis:**

➢ It involves examining and explaining the **components of budget** expenditure and revenue.

❖ **Risk Analysis:**

➢ Risk Analysis seeks to identify **measure and reduce various risk exposures** facing a projects.

▶ Project Evaluation > What does it involve?

Project evaluation typically analyzes various aspects:

- ❖ **Scope and Objectives:** Were all planned features delivered, and did they meet project goals?

- ❖ **Cost and Budget:** Did the project stay within budget, and were resources efficiently utilized?

- ❖ **Schedule and Timeline:** Were the project delivered on time, and were deadlines met?

- ❖ **Quality and Performance:** Do the software meet quality standards and perform as expected?

- ❖ **Stakeholder Satisfaction:** Were stakeholders satisfied with the project outcome and communication throughout?

- ❖ **Risk Management:** Were potential risks identified and mitigated effectively?

▶ **Project Evaluation** ➤ **Why is it important?**

❖ **Identify successes and failures:** Learn from what worked well and what didn't to improve future projects.

❖ **Improve processes and methodologies:** Use insights to refine your project management approach.

❖ **Demonstrate value and accountability:** Show stakeholders the project's impact and justify future investments.

❖ **Motivate and reward team members:** Recognize achievements and address areas for improvement.

▶ Project Evaluation > Benefits

❖ **Enhanced Decision-Making:** Helps in making informed decisions based on data and analysis.

❖ **Reduced Risk of Future Failures:** Identifies potential issues early, allowing for corrective actions.

❖ **Improved Project Management Practices:** Provides insights into what worked well and what didn't, helping refine practices.

❖ **Increased Stakeholder Confidence:** Demonstrates accountability and transparency, boosting stakeholder trust.

▶ **Project Evaluation** ➤ **Challenges:**

❖ **Defining Meaningful Metrics:** Ensuring metrics that are relevant and accurately reflect project performance.

❖ **Time and Resource Commitment:** Evaluation can be resource-intensive, requiring significant time and effort.

❖ **Potential for Bias:** Subjective judgments can influence evaluation outcomes.

► Project Evaluation

Ask the following **Wh-based Questions** in every topic while you create a software project or in any project management topic.

❖ Always perform **5W + 1H** questions and find its answers:

  ➢ What

  ➢ Where

  ➢ When

  ➢ Why

  ➢ Who

  ➢ How

▶ Project Evaluation

❖ **Who is responsible for Project Evaluation?**

➢ Senior Management

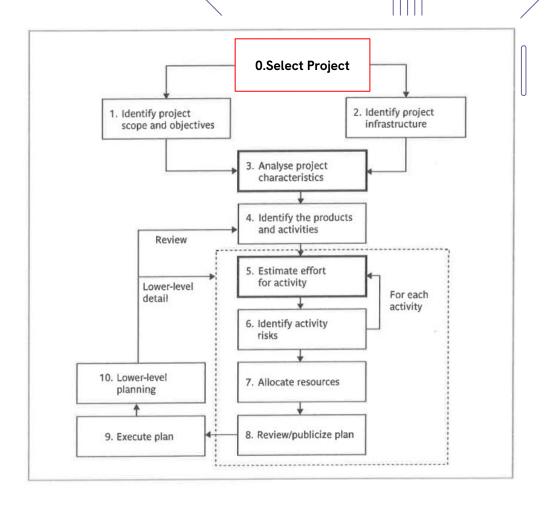➢ Project Manager/ Coordinator

➢ Team Leader

❖ **What assessment are performed for project evaluation/ accepting or rejecting a project?**

➢ Strategic Assessment

➢ Technical Assessment

➢ Economic Assessment { Cost-Benefit Analysis, Cash Flow Forecasting, Cost Benefit Evaluation Technique, Risk Evaluation}

When is Project Evaluation Done?

❖ Usually at the beginning of the project (Step 0 of the Step Wise Framework)

Strategic Assessment

4.1.1

BCA 7th Sem - Shishir Ghimire

▶ Strategic Assessment:

Strategic assessment is the critical **first step of evaluating potential projects** before diving into development. It is much like **checking a map and compass** before a journey to ensure the right direction and worthiness of the journey.

❖ Used to assess whether a **project fits in the long-term goal** of the organization.

❖ Usually **carried out by** senior management.

❖ Needs a **strategic plan** that clearly defines the objectives of the organization.

❖ It evaluates individual projects **against** the strategic plan or the overall business objectives.

❖ **Examples of Applications:**

➢ What projects do Facebook prefer vs. LinkedIn and why?

➢ Suitable for projects developed for use in the Organization Portfolio Management.

➢ Suitable for projects developed for other companies by software.

▶ **Strategic Assessment** ➤ **Types**:

❖ **Portfolio Management:**

➤ A portfolio is a **collection of projects and programs** that are managed as a group to achieve strategic objectives.

  ■ An organization may have **one portfolio**, which would then **consist of all projects**, programs, and operational work within the company.

  ■ It **may also establish several portfolios** for project selection and ongoing investment decisions.

➤ Suitable for projects **developed for other companies by software houses**.

➤ Provides an **overview of all the projects** that an organization is undertaking or is considering.

➤ It prioritizes the allocation of resources to projects and **decides** which new projects should be accepted and which existing ones should be dropped.

➤ **PMBOK:** "Projects, programs, other portfolios, and operations managed as a group to achieve strategic objectives."
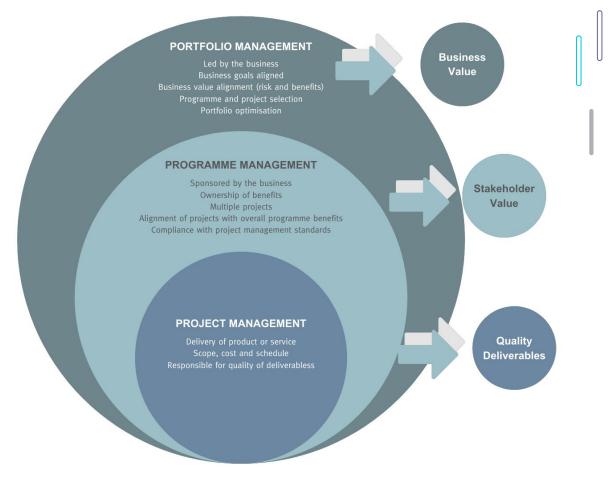
▶ **Strategic Assessment** ➤ **Types** :

❖ **Program Management:**

➤ In some cases, it's important that a group of projects is managed **in a coordinated way** to ensure that value is achieved.

➤ In project management terms, **this collection of projects becomes a program.** Like a project, a program is a temporary organization, so when the related projects are complete, the **program is complete.**

➤ Suitable for **projects developed for use in the organization**.

➤ **PMBOK**: "The application of knowledge and skills to achieve program objectives and to obtain benefits and **control not available by managing related program components individually.**"

➤ **Objectives:**

■ How does the project contribute to the long-term goal of the organization?

● For example, will the product increase the market share? By how much?

▶ **Strategic Assessment**
> Types :

PORTFOLIO MANAGEMENT
Led by the business
Business goals aligned
Business value alignment (risk and benefits)
Programme and project selection
Portfolio optimisation

PROGRAMME MANAGEMENT
Sponsored by the business
Ownership of benefits
Multiple projects
Alignment of projects with overall programme benefits
Compliance with project management standards

PROJECT MANAGEMENT
Delivery of product or service
Scope, cost and schedule
Responsible for quality of deliverabless

Business Value

Stakeholder Value

Quality Deliverables

Technical Assessment

4.1.2

BCA 7th Sem - Shishir Ghimire

▶ Technical Assessment:

Technical assessment evaluates the **technical feasibility and challenges of a project**, ensuring it can be successfully developed and implemented.

❖ **Characteristics:**

➢ **Feasibility Analysis:** Determines if the project is technically possible.

➢ **Identifies the technical resources** needed, including hardware, software, and technical skills.

➢ **Technical Risks:** Assesses potential technical risks and their impact on the project.

➢ **Compatibility:** Ensures compatibility with existing systems and technologies.

➢ **Compliance:** Checks for adherence to technical standards and regulatory compliance.

❖ **Examples of Applications:**

➢ **System Upgrades:** Assessing the feasibility and impact of upgrading existing systems.

➢ **New Technology Adoption:** Evaluating the technical challenges of integrating new technologies.

▶ Technical Assessment:

- ➢ **Software Development:** Analyzing the technical requirements and risks associated with developing new software applications.

- ❖ Technical assessments are carried out at **various stages** in software development:

  - ➢ **Project Initiation:** Conducted during feasibility studies to select the most suitable technology for the project.

  - ➢ **Design and Development:** Throughout the development process to identify and address technical challenges and ensure the project remains technically viable.

  - ➢ **Deployment and Testing:** To ensure the software performs as expected and meets all technical requirements before it goes live.

- ❖ **Who Conducts the Assessment?**

  - ➢ The assessment team may include **software architects, developers, technical leads, and sometimes external consultants** specializing in the chosen technology.

Economic & Risk Assessment

4.1.3

BCA 7th Sem - Shishir Ghimire

▶ Economic & Risk Assessment:

❖ Economic Analysis in project analysis involves **evaluating the financial feasibility** of a project by **analyzing its cost benefits and risks**.

❖ This includes **identifying and estimating the cost of resources** such as labour materials, equipment, revenue forecast and accessing the potential risks and benefits of the project.

❖ **Goal:**

➤ **Determine** whether a project is financially viable or not.

➤ Provide information that can be used to **make decisions** about the project allocation of resources.

▶ Economic & Risk Assessment:

❖ **How Economic and Risk Assessment are performed?**

➢ Cost-Based Analysis

➢ Cash-Flow Analysis

➢ Cost-Based Evaluation Technique

➢ Risk Analysis

Cost-Benefit Analysis

4.2

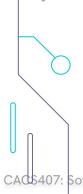BCA 7th Sem - Shishir Ghimire

▶ Cost-Benefit Analysis:

❖ Cost-benefit analysis is a simple technique for comparing the business value a project will produce with the cost of producing it.

❖ Project managers use cost-benefit analysis in the project initiation phase to show the value of doing a project.

❖ The cost-benefit analysis compares the project's costs to the business value it will deliver.

❖ CBA measures **costs and benefits to the community of adopting a particular course** of action e.g. Constructing a dam, by-pass etc.

❖ CBA is a **decision making device** for evaluating activities that are not priced by the market.

▶ Cost-Benefit Analysis > Purpose :

❖ There two main purposes in using CBA:

➢ To determine if the project is sound, justifiable and feasible by figuring out if its benefits outweigh costs.

➢ To offer a baseline for comparing projects by determining which project's benefits are greater than its costs

❖ Benefits of a Software Project:

➢ The benefits of a software project include four types:

■ Cost-savings benefits

■ Cost-avoidance benefits

■ Improved-service-level benefits
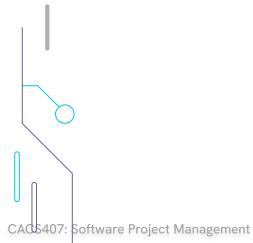
■ Improved Information benefits

▶ Cost-Benefit Analysis > Purpose :

❖ The cost of a software project include five types:

  ➢ Hardware Costs

  ➢ Personnel Costs

  ➢ Facility Costs

  ➢ Operating Costs

  ➢ Supply Costs

▶ **Cost-Benefit Analysis > Issues :**

❖   Is the project worthwhile financially?

❖   Is it the best option?

❖   Should it be undertaken at all?

▶ **Cost-Benefit Analysis** ➤ **Classification** :

❖ Costs and benefits can be classified as follows:

- ➤ (a) Tangible or Intangible Costs and Benefits
- ➤ (b) Fixed or Variable Costs and Benefits
- ➤ (c) Direct or Indirect Costs and Benefits

BCA 7th Sem - Shishir Ghimire

▶ **Cost-Benefit Analysis** ➤ **Classification** :

❖ **Tangible Cost:** An outlay of cash for any specific item or activity is referred to as a tangible cost. These costs are known and can be estimated quite accurately. Hardware costs, salaries for professionals, software cost are all tangible costs

❖ **Intangible Cost:** Costs that are known to exist but their financial value cannot be exactly measured are referred to as intangible costs. For example, employee morale problem because of installing new system is an intangible cost.

❖ **Direct Costs:** Direct costs are those which are directly associated with a system. Direct costs are having rupee value associated with it. They are-applied directly to the operator. For example, the purchase of pen drive for NRs.2, 000/- is a direct cost because we can associate the pen drive with money spent.

▶ Cost-Benefit Analysis ➤ Classification :

❖ **Indirect Costs:** Indirect costs result from the operations that are not directly associated with the system. They are often referred to as overhead expenses. For example, cost of space to install a system, insurance, maintenance of computer center, heat, light and air-conditioning are all indirect costs.

❖ **Fixed Costs:** Fixed costs are considered as sunk costs. They don't change. Once encountered, they will not recur. For example, the purchase of equipment for a computer center is called as fixed cost as it remains constant whether in equipment is being used extensively or not. Similarly, the insurance, and purchase of software, etc.

❖ **Variable Costs:** Variable costs are incurred on a regular basis. Recurring period may be weekly or monthly depending upon the system. They are generally proportional to world volume and continue as long as the system is in operation. For example, the cost of computer forms vary in proportion to the amount of processing or the length of the reports desired.

BCA 7th Sem - Shishir Ghimire

▶ Cost-Benefit Analysis ➢ Process :

Identification of costs & benefits → Evaluation of costs & benefits → Choice of System

❖ **Step-1: Identification of Costs & Benefits:** Identify the costs & benefits associated with a task / operation, or investment project.

❖ **Step-2: Evaluation of costs** (tangible, intangible, fixed, variable, direct, & indirect) & benefits associated with each task / operation / investment project.

❖ **Step-3: Choice of system:** Select the best system which involves least cost & yields maximum benefits.

▶ **Cost-Benefit Analysis Process ➤ Step-1 :**

- ❖ **Step-1:** Identifying and estimating all of the costs and benefits of carrying out the project and expressing these costs and benefits in common units. It includes

  - ➢ Development cost of system.

  - ➢ Operating cost of system.

  - ➢ Benefits obtained by system.

- ❖ When new system is developed by the proposed system, then new system should reflect the above three as same as proposed system.

- ❖ Example: sales order processing system which gives benefit due to use of new system.

▶ Cost-Benefit Analysis Process ➤ Step-2 :

❖ **Step-2:** Expressing these costs and benefits in common units. Calculates net benefit.

■ ***Net benefit = total benefit – total cost.*** (Cost should be expressed in monetary terms)

❖ **Three types of cost and benefits**

➢ **Development costs:** includes salary and other employment cost of staff involved.

➢ **Setup costs:** includes the cost of implementation of system such as hardware, and also file conversion, recruitment and staff training.

➢ **Operational cost:** cost require to operate system, after it is installed.

➢ **Direct benefits:** directly obtained benefit by making use of/operating the system. Example: reduction of salary bills, through the introduction of a new, computerized system.

➢ **Assessable indirect benefits (Secondary Benefits) :** these benefits are obtained due to updating / upgrading the performance of current system. Example: "use of user – friendly screen", which promotes reduction in errors, thus increases the benefit.

➢ **Intangible benefits (Indirect Benefits) :** these benefits are longer term, difficult to quantify. Example: enhanced job interest leads reduction of staff turnover, in turn leads lower recruitment costs.

▶ Cost-Benefit Analysis Process > Exercise :

Calculate net profit for the following

Year	Project 1	Project 2	Project 3
0	-100000	-1000000	-120000
1	10000	200000	30000
2	10000	200000	30000
3	10000	200000	30000
4	20000	200000	30000
5	100000	300000	75000

BCA 7th Sem - Shishir Ghimire

▶ Cost-Benefit Analysis Process ➤ Exercise :

Net profit=total costs-total incomes

Calculate net profit for the following

Year	Project 1	Project 2	Project 3
0	-100000	-1000000	-120000
1	10000	200000	30000
2	10000	200000	30000
3	10000	200000	30000
4	20000	200000	30000
5	100000	300000	75000

Solution

Year	Project 1	Project 2	Project 3
0	-100000	-1000000	-120000
1	10000	200000	30000
2	10000	200000	30000
3	10000	200000	30000
4	20000	200000	30000
5	100000	300000	75000
Net Profit	50000	100000	75000

Cash Flow Forecasting

4.3

▶ Cash Flow Forecasting:

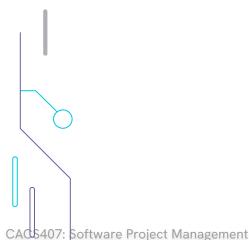❖ A cash flow forecast indicates **when expenditure and income will take place**.

Typically products generate a negative cash flow during their development followed by a positive cash flow over their operating life. There might be decommissioning costs at the end of a product's life

Figure *Typical product life cycle cash flow.*

❖ It is vital to have some forecast of when expenditure such as the payment of salaries and bank interest will take place and when any income is to be expected, such as payment on completion or possibly stage payments.

▶ Cash Flow Forecasting:

❖ **Accurate** cash flow **forecasting is not easy** as it generally **needs to be done early in the project s 's life cycle.**

❖ While estimating future cash flows, it is usual to **ignore the effects of inflation** as this **increases the uncertainty of forecasts.**

❖

▶ Cash Flow Forecasting:

Table A: *Four project cash flow projections – figures are end of year totals (£)*

Year	Project 1	Project 2	Project 3	Project 4
0	−100,000	−1,000,000	−100,000	−120,000
1	10,000	200,000	30,000	30,000
2	10,000	200,000	30,000	30,000
3	10,000	200,000	30,000	30,000
4	20,000	200,000	30,000	30,000
5	100,000	300,000	30,000	75,000
Net profit	50,000	100,000	50,000	75,000

Cash flows take place at the end of each year. The year 0 figure represents the initial investment made at the start of the project.

❖ The table illustrates cash flow forecasts for four projects. **In each case it is assumed that the cash flows take place at end of each year.**

❖ For short term projects or where candidate projects demonstrate significant seasonal cash flow patterns, it can be advisable to produce quarterly or even monthly cash flow forecasts.

▶ Cash Flow Forecasting:

❖ **If We Could Get It Right**

- ➤ **Liquidity Management:**
    - ■ **Having Funds Available To Meet All Known and Unknown Commitment**
- ➤ **Minimise Cost of Funds**
- ➤ **Maximise Interest Earnings**
- ➤ **Budgeting and Control**
- ➤ **Currency Risk Management**
- ➤ **Working Capital Management**

Cost Benefit Evaluation Technique
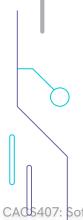
4.4

▶ Cost Benefit Evaluation Technique:

❖ We would consider proceeding with a project only where the benefits outweigh the cost.

❖ The timing of costs and benefits (how many years???)

❖ Benefits relative to the size of investment.

▶ Cost Benefit Evaluation Technique:

❖ **Payback Period:** This calculates **how long** it takes for the **project's benefits to "pay back" the initial investment.** It's easy to understand but **doesn't consider** the time value of money or ongoing benefits beyond the payback period.

❖ **Return on Investment (ROI):** This expresses the **project's profit as a percentage of the initial investment**. It provides a broader picture than the payback period but can be skewed by how you define benefits and costs.

❖ **Net Present Value (NPV):** This takes into account the **time value of money**, discounting future benefits back to their present value. It's considered a more accurate analysis for long-term projects but requires more complex calculations.

▶ **Cost Benefit Evaluation Technique:**

❖ **Internal Rate of Return (IRR):** This calculates the **discount rate at which the NPV becomes zero**. It essentially tells you the minimum acceptable return on investment for the project to be considered worthwhile.

❖ **Cost-Effectiveness Analysis:** This focuses on **comparing the costs of alternative options** that achieve the same goal. It's helpful when different projects offer similar benefits but vary in cost.

❖ **Break-Even Analysis:** This calculates the point where the **project's costs and benefits equal each other**. It helps you understand the minimum level of sales or usage needed to cover development costs.

▶ Cost Benefit Evaluation Technique> Net Profit:

❖ The net profit of a project is the **difference between the total costs and the total income** over the life of the project.

❖ It shows what the company has earned (or lost) in a given period of time (usually one year).

❖ It is also called **net income** or net earnings.

❖ **Formula:**

  ➢ *Net profit= Total costs - Total incomes*

  ➢ **(Numerical for Net Profit Already Done in Above Section)**

▶ CBET > Pay Back Period :

❖ **Payback period:** The payback period is the time taken to **break even or pay back** the initial investment.

❖ Normally, the project with the **shortest payback period** will be chosen on the basis that an organization will wish to minimize the time that a project is '**in debt**'.

❖ **Advantages**

  ➢ simple and easy to calculate.

❖ **Disadvantages**

  ➢ It **attaches no value** to cash flows after the end of the payback period.

  ➢ It makes no adjustments for risk.

  ➢ It is not directly related to **wealth maximization** as NPV is.

  ➢ It ignores the time value of money.

  ➢ The "cut off" period is arbitrary.

▶ CBET ➤ Pay Back Period :

❖ **Formula to Calculate:**

$$\text{Payback Period} = \frac{\text{Initial Investment}}{\text{Annual Cash Inflow}}$$

Example Calculation:

- Initial Investment: $50,000

- Annual Cash Inflow: $10,000

$$\text{Payback Period} = \frac{50,000}{10,000} = 5\text{ years}$$

▶ CBET ➤ Return on Investment :

❖ The return on investment (ROI), also known as the accounting rate of return (ARR), provides **a way of comparing the net profitability to the investment required**. It is expressed as a percentage.

- A straight forward common formula for calculation is

$$ROI = \frac{AverageAnnualprofit}{TotalInvestment} \times 100$$

Average Annual Profit = $\frac{\text{Net Profit}}{\text{Total no. of Years}}$

Example Calculation:

- **Net Benefit:** $15,000

- **Cost of Investment:** $50,000

$$ROI = \left(\frac{15,000}{50,000}\right) \times 100 = 30\%$$

BCA 7th Sem - Shishir Ghimire

▶ CBET ➤ Return on Investment :

Calculate ROI for following.

Year	Project 1
0	-100000
1	10000
2	10000
3	10000
4	20000
5	100000
Net Profit	50000

Answer: Total Investment   = 1,00,000
Net Profit   = 50,000
Total no. of Year   = 5
Average Annual Profit   = 50,000/5
   = NRs. 10,000

ROI   = (10,000/1,00,000) *100
   = 10%

Note: The higher the ROI, the better.

▶ CBET ➤ Return on Investment :

❖ Calculate the ROI for following project (5).

Year	Project Cash Flow
0	-50,000
1	5,000
2	5,000
3	20,000
4	40,000
5	50,000

BCA 7th Sem - Shishir Ghimire

▶ CBET ➤ Net Present Value (NPV) :

❖ **Net present value (NPV):** It is the sum of the present values of all future amounts. Present value is the value which a **future amount** is worth at present. It **takes into account the profitability of a project and the timing of the cash flows**.

❖ **Discount rate:** Is the annual rate by which we discount future earnings. e.g. If discount rate is 10% and the return of an investment in a year is $110, the present value of the investment is $100.

❖ The present value of any future cash flow may be obtained by using the following formula.

$$presentValue = \frac{ValueInYearT}{(1+r)^T} \qquad DiscountFactor = \frac{1}{(1+r)^T}$$

- Where
- r = discount rate expressedd as a decimal value
- T = is the number of years into the future that the cash flow occurs.

▶ **CBET** ➤ **Net Present Value (NPV)** :

❖ **Disadvantage:**

➢ May not be **directly comparable** with earnings from other investments or the costs of borrowing capital.

❖ **Issues in NPV**

➢ Choosing an appropriate discount rate is difficult

➢ Ensuring that the rankings of projects are **not sensitive to small changes** in discount rate

❖ **NPV Rule:**

■ If **NPV > 0**, accept the Project.

■ If **NPV = 0**, accept or reject the project.

■ If **NPV < 0**, reject the project.

▶ CBET ➤ Net Present Value (NPV) :

Formula to Calculate:

$$NPV = \sum_{t=1}^{n} \frac{R_t}{(1+i)^t} - C_0$$

Where:

- R_t = Net cash inflow at time t
- i = Discount rate
- C_0 = Initial investment

Example Calculation:

- **Cash Flows:** Year 1 = $10,000, Year 2 = $15,000
- **Initial Investment:** $20,000
- **Discount Rate:** 10%

$$NPV = \frac{10,000}{(1+0.1)^1} + \frac{15,000}{(1+0.1)^2} - 20,000$$

$$NPV = 9,090.91 + 12,396.69 - 20,000 = 1,487.60$$

▶ CBET ➤ Future Worth :

❖ Future worth refers to the value of an investment **at a specific point in the future**, taking into account factors such as interest rates and price rise. It is a financial concept that is used to determine the **present value of future cash flows or streams of cash flows.** It is used in financial analysis and decision-making regarding different investment options and to evaluate the potential return on investment.

❖ There are two types of future value calculations:

➤ **1. Future Value of a Lump Sum:**

  ■ It is the value of a **single deposit**, like a bank CD over time.

$$FV = PV \times (1 + r)^n$$

Where:

- FV: Future Value
- PV: Deposit or Present Value
- r: Rate of interest over a period of time
- n: Number of time periods such as years

▶ CBET ➤ Future Worth :

➤ 2. **Future Value of an Annuity:**

   ■ It is the value of a **series of payments**.

Formula:

$$FV = PMT \times \left[\frac{(1+r)^n - 1}{r} \right]$$

Where:

- PMT: Payment or contribution

- r: Rate of interest over a period of time

- n: Number of time periods

▶ CBET ❯ Internal Rate of Return (IRR) :

❖ IRR is the **discount rate** at which the **NPV of a project becomes zero**. It represents the break-even cost of capital for a project.

❖ One disadvantage of NPV is, it might not be directly comparable with earnings from other investments or the costs of borrowing capital. Such costs are usually quoted as a percentage of interest rate.

❖ The internal rate of return attempts to provide a **profitability measure as a percentage return** that is directly comparable with interest rates.

❖ Thus, a project that showed estimated IRR of 10% would be worthwhile if the capital could be borrowed for less than 10% or of the capital could not be invested elsewhere for a return greater than 10%

❖ The IRR is calculated as that percentage discount rate that would produce an **NPV of zero**.

❖ It is most easily calculated using a spreadsheet that provides functions for calculating IRR.

▶ CBET ➤ Internal Rate of Return (IRR) :

❖ **IRR is calculated using the formula :** (CF stands for Cash Flow and r is the internal rate of return)

$$\left[\frac{CF_1}{(1+r)^1} + \frac{CF_2}{(1+r)^2} + \frac{CF_3}{(1+r)^3} + ...\right] - \text{Initial Investment} = 0$$

❖ **Decision Rule:** Accept the project if the IRR is greater than the required return

➤ **Internal Rate of Return Rules:** In IRR decisions, if we have only one project, most of the time we need the basic rule «independent project»:

  ■ IRR > Cost of capital (should be accepted)

  ■ IRR = Cost of capital (provides the minimum return)

  ■ IRR < Cost of capital (shouldn't be accepted)

▶ CBET ➤ Benefit-Cost Ratio :

❖ BCR is the ratio of the **present value of benefits** to the **present value of costs**.

❖ Benefit-cost ratio (BCR) analysis is a method used to determine the economic feasibility of a project or investment. It involves **comparing** the total benefits of a project to its total costs and expressing the result as a ratio.

➢ A **BCR greater than 1** indicates that the project is a good investment.

➢ A **BCR less than 1** indicates that the project is a poor investment.

❖ It determines the economic value of a project.

$$\text{Benefit-Cost Ratio} = \frac{\text{PV of Benefits Expected from the Project}}{\text{PV of the Cost of the Project}}$$

▶ CBET ➤ Benefit-Cost Ratio :

Formula to Calculate:

$$BCR = \frac{\text{Present Value of Benefits}}{\text{Present Value of Costs}}$$
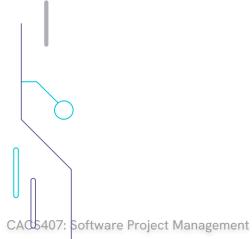
Example Calculation:

- Present Value of Benefits: $30,000

- Present Value of Costs: $20,000

$$BCR = \frac{30,000}{20,000} = 1.5$$

BCA 7th Sem - Shishir Ghimire

▶ **CBET** ➤ To Read Topics :

- ❖ CV (Cost Variance)

- ❖ EV (Earned Value)

- ❖ Schedule Performance Index (SPI)

- ❖ Cost Performance Index (CPI)

Risk Evaluation 4.5

▶ Risk Evaluation:

❖ Risk evaluation in software project management is the essential process of **identifying, analyzing**, and **assessing potential risks** that could **negatively impact** your project's success. It's like looking at a weather forecast before starting a journey - you want to anticipate potential storms and be prepared to navigate them.

❖ Due to these factors, every software development project carries elements of **uncertainty**, known as project risks.

❖ The **success** of such projects heavily **depends on how** these risks are identified, assessed, prioritized, and managed.

❖ Simply being aware of risks is **not sufficient;** project managers must actively **evaluate and mitigate** them to achieve successful outcomes.

❖ Risk evaluation determines whether to proceed with a project and assesses whether the project meets its objectives. Risks may arise when **projects deviate from their original specifications** or fail to achieve desired outcomes.

▶ **Key Aspects of Risk Evaluation:**

1. Risk Identification and Ranking

2. Risk and NPV

3. Cost-Benefit Analysis

4. Risk Profile Analysis

5. Using Decision Tree

▶ Risk Identification and Ranking:

❖ In any project evaluation we should attempt to **identify the risks** and **quantify their potential effects**.

❖ A common approach involves **constructing a project risk matrix** using a checklist of potential risks. Each risk is **classified** based on:

- ■ Relative importance
- ■ Likelihood of occurrence

❖ Serious but **unlikely risks may require less attention** than less severe risks with higher probabilities.

A fragment of a basic project risk matrix

Risk	Importance	Likelihood
Software never completed or delivered	H	—
Project cancelled after design stage	H	—
Software delivered late	M	M
Development budget exceeded ≤ 20%	L	M
Development budget exceeded > 20%	M	L
Maintenance costs higher than estimated	L	L
Response time targets not met	L	H

BCA 7th Sem - Shishir Ghimire

▶ Risk and Net Present Value (NPV):

❖ Where a project is **relatively risky** it is common practice to use a **higher discount rate** to calculate NPV.

❖ This addition or risk premium, might for example, be an **additional 2%** for a reasonably safe project or **5%** for a fairly risky one.

   ■ **Low risk:** +2%

   ■ **High risk:** +5%

❖ Projects may be categorized as **high, medium or low risk** using a scoring method and risk premiums designated for each category, providing a **structured way** to account for risk.

▶ Cost-Benefit Analysis:

❖ A more sophisticated evaluation method **involves:**

➢ Estimating **possible outcomes** and their probabilities.

➢ Creating **multiple cash flow forecasts** with associated probabilities.

➢ Calculating the **project's value** by summing weighted costs or benefits.

❖ This provides a probabilistic view of project outcomes.

▶ Risk Profile Analysis:

❖ **Sensitivity analysis** evaluates how changes in project parameters affect profitability.

➢ Each **parameter is adjusted** (e.g., ±5%), and **expected costs/benefits** are recalculated.

➢ Sensitivity highlights **factors most critical** to the project's success.

❖ **Actions include:**

➢ Exercising **greater control** over critical factors.

➢ **Mitigating** adverse effects.

➢ Abandoning the project if **risks are too high** to manage effectively.

❖ Monte Carlo techniques can be used for **evaluating combinations of circumstances**, offering a **more comprehensive approach** than traditional sensitivity analysis.

▶ Using Decision Trees:

❖ Decision trees provide a **structured way** to evaluate risks and their impact on future outcomes.

❖ **This approach allows:**

- Identifying significant risks.

- Choosing suitable courses of action.

❖ Decision trees can illustrate the **potential profitability of different scenarios** and guide decisions that impact future project options.

▶ Benefits of Risk Evaluation:

❖ **Improved project outcomes:** By proactively addressing risks, you increase your chances of staying on track and meeting your goals.

❖ **More informed decision-making:** Understanding potential risks helps you make better choices about resource allocation, scheduling, and project scope.

❖ **Enhanced communication and collaboration:** Openly discussing risks fosters transparency and helps team members work together to mitigate them.

❖ **Reduced stress and anxiety:** Knowing you've prepared for potential challenges can improve team morale and productivity.

Selection of an Appropriate Project Report

4.6

BCA 7th Sem - Shishir Ghimire

▶ Selecting the Appropriate Report:

Factors to Consider:

❖ **Purpose of the Report:** Are you providing a status update, seeking funding approval, analyzing risks, or presenting results? Different purposes require different information and formats.

❖ **Target Audience:** Who will be reading the report? Tailor the content and complexity to their level of understanding and interest.

❖ **Project Stage:** Is the project in the planning, development, testing, or deployment phase? Each stage requires different information to be reported.

❖ **Project Methodology:** Are you using agile, waterfall, or another methodology? Different methodologies use different reporting formats and metrics.

❖ **Company Standards:** Does your company have specific reporting templates or guidelines? Ensure your report adheres to these.

▶ **Selecting the Appropriate Report>** **Common Report** :

Here are some common reports used in software project management and their typical uses:

❖ **Project Status Report:** Provides a high-level overview of project progress, milestones achieved, challenges faced, and upcoming tasks.

❖ **Project Health Report:** Dives deeper into project health aspects like budget adherence, schedule compliance, team morale, and risk mitigation measures.

❖ **Project Risk Report:** Identifies and analyzes potential risks, their likelihood and impact, and proposed mitigation strategies.

▶ Selecting the Appropriate Report> Common Report :

❖ **Change Request Report:** Documents proposed changes to the project scope, requirements, or timeline, along with justifications and potential impacts.

❖ **Cost-Benefit Analysis Report:** Analyzes the projected costs and benefits of a project to assess its financial viability.

❖ **Test Summary Report:** Summarizes the test results, identifies bugs or defects, and outlines their severity and resolution plans.

❖ **Project Closure Report:** Documents the project's completion, achievements, lessons learned, and final deliverables

Project Approach:
Choosing Technologies,
Choice of Process Models,
Structured Methods

4.7

BCA 7th Sem - Shishir Ghimire

▶ Project Approach:

❖ A Project Approach in software project management defines the overall set of principles methodologies and practices used to guide the planning,, execution and control of a software project.

❖ It's like choosing the blueprint and tools for building your software project.

❖ **Key Aspects of a Project Approach:**

❖ **Methodology:**

➢ Identifies the overarching framework for managing the project, such as waterfall, agile, Scrum, Kanban, Lean, or a hybrid approach. Each methodology has its own lifecycle, deliverables, and emphasis on areas like collaboration and flexibility.

▶ Project Approach:

❖ **Practices:**

➢ Specific techniques and tools used within the chosen methodology. For example, agile projects might use Kanban boards and daily stand-up meetings, while waterfall projects might rely on Gantt charts and formal change management processes.

❖ **Governance:**

➢ Defines the decision-making structure, roles, and responsibilities within the project. It clarifies who makes key decisions, how stakeholders are involved, and how communication flows.

❖ **Customization:**

➢ While methodologies offer frameworks, successful projects often adapt and tailor the approach to their specific needs, context, and team preferences.

Choosing Technologies

4.8

BCA 7th Sem - Shishir Ghimire

▶ Choosing Technologies:

❖ An outcome of project analysis will be the selection of the most appropriate methodologies and technologies

❖ Methodologies include techniques like various flavours of object-oriented development or SSADM.

❖ The chosen technology will influence the following aspects of a project.

➢ The training requirement for development staff;

➢ The types of staff to be recruited;

➢ The development environment – both hardware and software;

➢ System maintenance arrangement

BCA 7th Sem - Shishir Ghimire

▶ **Choosing Technologies** ➤ **Analyze Project Characteristics:**

- ❖ Is a data oriented or process oriented system to be implemented?

- ❖ Will the software that is to be produced be a general package or application specific?

- ❖ Is the system to be implemented of a particular type for which specific tools have been developed?

- ❖ Is the system to be created safety critical?

- ❖ What is the nature of the hardware/software environment in which the system will operate?

▶ **Choosing Technologies** ➤ **Analyze Project Characteristics:**

How would you categorize each of the following systems according to the classification above?

(a) a payroll system
(b) a system to control a bottling plant
(c) a system that holds details of the plans of plant used by a water company to supply water to consumers
(d) a software application to support project managers
(e) a system used by lawyers to get hold of case law relating to company taxation.

- ❖ A payroll system is a data oriented or information system that is application specific.

- ❖ The bottling plant system is an industrial system which contains embedded software.

- ❖ It is an IS that will make heavy use of computer graphics. The plant itself might use control software which might be safety critical.

- ❖ Project management software tools are often categorized as general packages. There would be a considerable information systems element to them.

- ❖ This could be an information retrieval package that is a general package. It is also a strong candidate for a knowledge based system.

▶ **Choosing Technologies > Analyze High-Level Project Risks:**

❖ **Uncertainty >>>>** the greater the uncertainties at the beginning of the project, the greater the risk that the project will be unsuccessful.

❖ **Product Uncertainty>>>>**

➤ How well the requirements are understood?

➤ The users themselves could be uncertain about what a proposed Information System is to do.
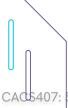
❖ **Process Uncertainty>>>>**

➤ It might be that the project under consideration is the first where an organization has tried to use a method that is new to them.

➤ Any changes in the way that the systems are developed is going to introduce uncertainty.

❖ **Resource Uncertainty>>>>**

➤ Main issue----unavailability of staff of the right ability and experience.

➤ The larger the number of resources need or the longer the duration of the project, the more inherently risky it is likely to be.

Choice of Process Model

4.9

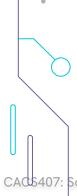BCA 7th Sem - Shishir Ghimire

▶ Choice of Process Models:

❖ There are different project management methodologies that cater to the needs of different projects spanned across different business domains.

❖ **Goals:** Provide Guidance for controlling and coordinating the tasks to achieve the end product and objective as effectively as possible

❖ There are different software processes but all involve:

➢ **Specification:** Defining what the system should do.

➢ **Design and Implementation:** Defining the organization of the system and implementing the system.

➢ **Validation:** Checking that it does what the customer wants.

➢ **Evolution:** Changing the system in response to changing customer needs.

▶ Choice of Process Models:

❖ Choosing the right process model is crucial for setting the groundwork for success. It's like selecting a roadmap for your project, outlining the stages activities and decision points encountered. Each process models has it's own strengths and weakness and the best fit depends on project needs and context.
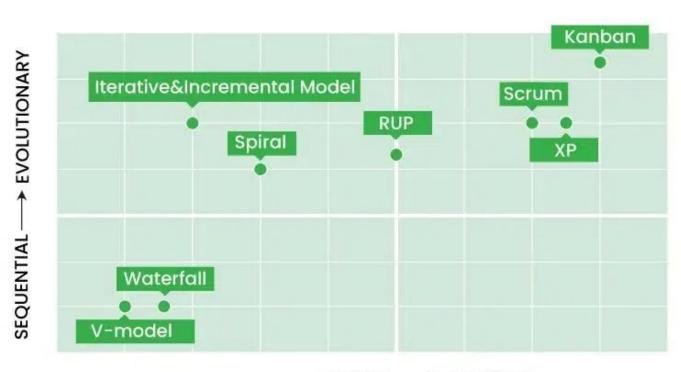
❖ **Different Types of Process Models:**

➢ Waterfall Model

➢ Incremental Model

➢ RAD Model

➢ Agile Model

➢ Iterative Model

➢ Spiral Model

➢ Prototype Model

Structured Methods

4.10

BCA 7th Sem - Shishir Ghimire

▶ Structured Methods:

BCA 7th Sem - Shishir Ghimire

▶ Structured Methods:

❖ Structured methods consist of sets of steps and rules which, when applied, generate system products such as use case diagrams. Each of these products is carefully defined. Such methods are more time consuming and expensive than more intuitive approaches. Structured methods refer to a collection of formal, step-by-step approaches used to develop and manage software projects.

❖ These methods aim to bring control, organization, and consistency to the development process, ensuring projects are delivered on time, within budget, and meet specified requirements.

❖ A structured method includes a design process model, notations to represent the design, report formats, rules, and design guidelines.

▶ Structured Methods:

❖   Structured methods may support some or all of the following models of a system:

➢   An **object model** that shows the object classes used in the system and their dependencies.

➢   A **sequence model** that shows how objects in the system interact when the system is executing.

➢   A **state transition model** that shows system states and the triggers for the transitions from one state to another.

➢   A **structural model** where the system components and their aggregations are documented.

➢   A **data flow model** where the system is modelled using the data transformations that take place as it is processed. This is not normally used in object-oriented methods but is still frequently used in real-time and business system design.

➢   A **use-case model** that shows the interactions between users and the system.

▶ **Structured Methods >** Key Characteristics :

❖ **Formal and well-defined:** They define clear phases, activities, and deliverables for each stage of the development process.

❖ **Emphasis on documentation:** Extensive documentation is created throughout the project, capturing requirements, design decisions, and test results.

❖ **Structured tools and techniques:** Specific tools and techniques are recommended for various tasks, like data flow diagrams for data analysis and entity-relationship diagrams for modeling data relationships.

❖ **Focus on quality control:** Quality assurance activities are integrated throughout the development process to identify and address issues early on.

▶ Structured Methods > Types :

❖ **Waterfall:** A sequential approach where each phase (requirements, design, development, testing, deployment) is completed before moving on to the next.

❖ **Spiral:** Combines features of waterfall and iterative models, allowing for risk assessment and prototyping before full commitment to each phase.

❖ **Prototyping:** Develops simplified versions of the software to gather user feedback and refine requirements before full development.

❖ **Agile:** Employs short iterations (sprints) with continuous delivery and feedback loops, adapting to changing requirements as the project progresses.

BCA 7th Sem - Shishir Ghimire

▶ **Structured Methods** ❯ **Benefits** :

❖ **Reduced risk:** Formal processes help identify and mitigate potential problems early on.

❖ **Improved project control:** Clear phases and deliverables aid in tracking progress and managing resources effectively.

❖ **Enhanced communication:** Detailed documentation facilitates communication between stakeholders and team members.

❖ **Better quality:** Emphasis on quality control ensures higher quality software is delivered.

▶ **Structured Methods** ＞ **Challenges** :

- ❖ **Can be inflexible:** May not adapt well to rapid changes in requirements or technology.

- ❖ **Time-consuming:** Extensive documentation can be resource-intensive and time-consuming.

- ❖ **Bureaucratic:** Strict adherence to processes might hinder creativity and innovation.

▶ **Waterfall Model** 4.10.1
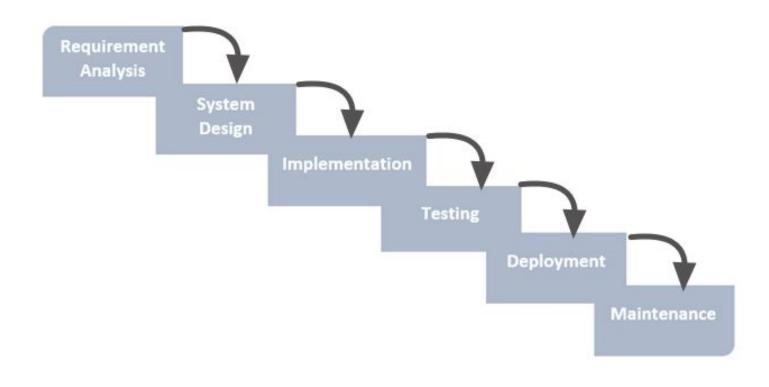
▶ Waterfall Model:

❖ **Definition**

➢ The Waterfall Model is a sequential software development process where progress flows downwards through distinct phases, like a waterfall. It is one of the oldest and most traditional methodologies used in software development.

❖ **Characteristics**

➢ Linear and sequential flow.

➢ Each phase must be completed before moving to the next.

➢ Well-documented process and deliverables.

➢ Emphasis on early-stage documentation.

BCA 7th Sem - Shishir Ghimire

▶ Waterfall Model:

▶ Waterfall Model:

❖ **Phases/Steps of the Model**

➢ **Requirement Analysis:** Gather and document all system and software requirements.

➢ **System Design:** Translate requirements into system and software designs.

➢ **Implementation:** Develop the code based on design documents.

➢ **Integration and Testing:** Integrate the components and test the system for defects.

➢ **Deployment:** Deploy the system for operational use.

➢ **Maintenance:** Address post-deployment issues, updates, and maintenance tasks.

▶ Waterfall Model:

❖ **Benefits of the Model**

➤ **Simple and Easy to Use:** Straightforward to understand and manage.

➤ **Well-Structured:** Phases are clearly defined.

➤ **Early Detection of Requirements Issues:** Problems in requirements are detected early.

➤ **Good for Small Projects:** Effective for small, well-understood projects.

❖ **Drawbacks of the Model**

➤ **Rigid:** No flexibility to make changes in between phases.

➤ **Late Testing:** Errors are detected only after the implementation phase.

➤ **Not Ideal for Complex or Large Projects:** Unsuitable for projects where requirements evolve over time.

➤ **High Risk:** Late discovery of errors can lead to significant rework.

▶ Waterfall Model:

❖ **When to Use:**

➢ Projects with clear, stable, and well-documented requirements.

➢ Projects where the technology stack is well-understood.

❖ **Example Cases:**

➢ Development of a payroll system for a small organization.

➢ Building a static website where the scope is fixed.

▶ **Incremental Model** 4.9.2

BCA 7th Sem - Shishir Ghimire
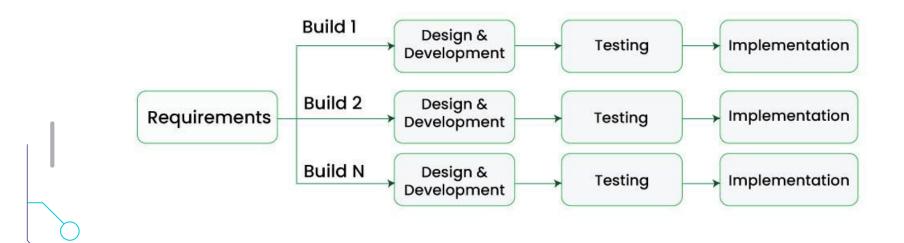
▶ Incremental Model:

❖ **Definition**

➢ The Incremental Model is a software development approach where the system is developed in small, manageable increments. Each increment builds on the previous one and delivers part of the functionality, eventually forming the complete system.

❖ **Characteristics**

➢ Combines iterative and sequential approaches.

➢ Delivers a working product after every increment.

➢ Changes can be incorporated between increments.

➢ User feedback is utilized in subsequent increments.

▶ Incremental Model:

BCA 7th Sem - Shishir Ghimire

▶ Incremental Model:

❖ **Phases/Steps of the Model**

➢ **Initial Planning:** Define overall system requirements and architecture.

➢ **Incremental Design:** Design for the current increment based on requirements.

➢ **Incremental Implementation**: Develop and deliver a working product increment.

➢ **Testing:** Test each increment individually and in integration with previous increments.

➢ **Integration and Deployment:** Combine all increments to build the final system.

▶ Incremental Model:

❖ **Benefits of the Model**

➢ Early Delivery: Early increments provide working functionality.

➢ Flexibility: Allows for changes based on feedback.

➢ Risk Management: Risks are identified and mitigated incrementally.

➢ Customer Satisfaction: Customers can see progress and provide input.

❖ **Drawbacks of the Model**

➢ Overlapping Phases: Increments can cause confusion if overlapping tasks are not managed well.

➢ Dependency Management: Changes in requirements can complicate the integration of new increments.

➢ Not Ideal for Small Projects: Overhead may not justify its use in simple systems.

BCA 7th Sem - Shishir Ghimire

▶ Incremental Model:

❖ **When to Use:**

➢ Projects with clear overall requirements but details may evolve.

➢ When early delivery of part of the functionality is required.

❖ **Example Cases:**

➢ Development of an online banking application with modules like login, account management, and fund transfers delivered incrementally.

➢ Creating a learning management system (LMS) where course creation, enrollment, and grading features are implemented in phases.
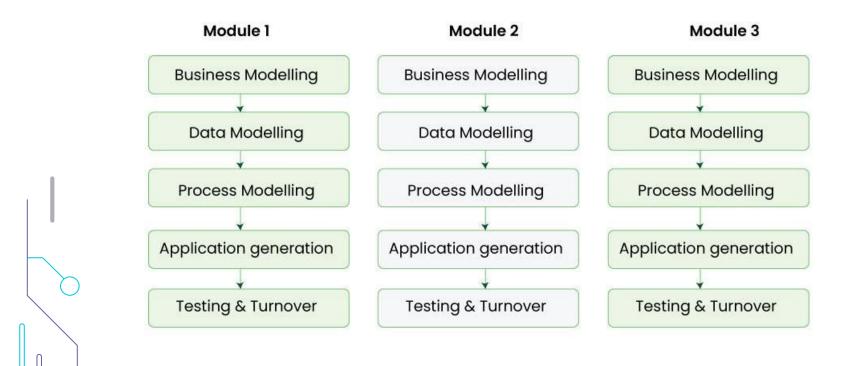
Rapid Application Development (RAD) Model

4.9.3

▶ Rapid Application Development (RAD) Model:

❖ **Definition**

➢ The RAD Model is a software development approach that emphasizes quick development and iteration through user feedback and prototyping. It prioritizes rapid delivery over extensive planning.

❖ **Characteristics**

➢ Heavy focus on user involvement.

➢ Relies on component-based development and prototyping.

➢ Works best with skilled developers and efficient tools.

▶ Rapid Application Development (RAD) Model:

▶ Rapid Application Development (RAD) Model:

❖ **Phases/Steps of the Model**

➢ **Business Modeling:** Understand business objectives and workflows.

➢ **Data Modeling:** Define and refine the data required for business processes.

➢ **Process Modeling:** Outline how data is transformed through workflows.

➢ **Application Generation:** Rapidly develop prototypes using tools and frameworks.

➢ **Testing and Turnover:** Conduct testing and deploy the final product.

▶ Rapid Application Development (RAD) Model:

❖ **Benefits of the Model**

➢ **Faster Development:** Reduces development time significantly.

➢ **User Feedback:** Frequent feedback ensures alignment with user needs.

➢ **Flexibility:** Easy to incorporate changes based on feedback.

➢ **Risk Reduction:** Issues are identified early during prototyping.

❖ **Drawbacks of the Model**

➢ **Requires Skilled Team:** Needs experienced developers and designers.

➢ **High Resource Requirement:** Demands efficient tools and user involvement.

➢ **Not Suitable for Complex Systems:** Lacks emphasis on scalability for large projects.

BCA 7th Sem - Shishir Ghimire

▶ Rapid Application Development (RAD) Model:

❖ **When to Use:**

- ➢ When requirements are clear but can evolve during development.
- ➢ For small to medium-sized projects needing quick delivery.

❖ **Example Cases:**

- ➢ Developing a mobile app for event ticket booking.
- ➢ Building a prototype for a customer relationship management (CRM) system.

▶ **Agile Model**

4.9.4

▶ Agile Model:

❖ **Definition**

➢ The Agile Model is an iterative and incremental software development methodology focusing on collaboration, customer feedback, and rapid releases of small functional parts of the product.

❖ **Characteristics**

➢ Iterative and adaptive process.

➢ Emphasizes collaboration and customer involvement.

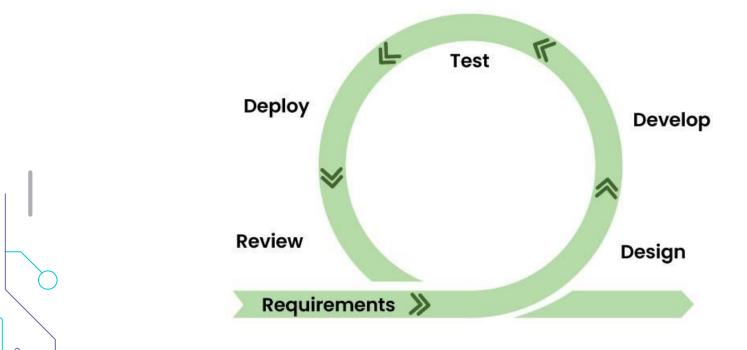➢ Focus on delivering working software quickly.

▶ Agile Model:

❖ **Phases/Steps of the Model**

➤ **Requirements Gathering:** Define project requirements, evaluate business opportunities, and assess technical and economic feasibility.

➤ **Design the Requirements:** Collaborate with stakeholders to define requirements using user flow or UML diagrams.

➤ **Construction/Iteration:** Developers and designers build the product incrementally, starting with minimal functionality and improving it over iterations.

➤ **Testing:** The Quality Assurance team tests the product for bugs and performance issues.

➤ **Deployment:** The product is released to the user's environment.

➤ **Feedback:** Gather user feedback post-release to make improvements for future iterations.

▶ Agile Model:

▶ Agile Model:

❖ **Benefits of the Model**

  ➢ Flexibility: Easily accommodates changing requirements.

  ➢ Customer Satisfaction: Frequent deliveries ensure alignment with user expectations.

  ➢ Risk Reduction: Continuous iterations minimize risks.

  ➢ Improved Collaboration: Encourages teamwork and communication.

❖ **Drawbacks of the Model**

  ➢ Requires Skilled Team: Demands highly skilled and cooperative teams.

  ➢ Documentation Overlooked: Focus on rapid development may reduce documentation.

  ➢ Not Ideal for Fixed-Scope Projects: Continuous changes can lead to scope creep.

▶ Agile Model:

❖ **When to Use:**

➢ Projects with evolving requirements and high customer involvement.

➢ Complex projects requiring frequent releases.

❖ **Example Cases:**

➢ Development of a ride-sharing application.

➢ Building an e-commerce platform with incremental feature additions.

▶ **Iterative Model**   **4.9.5**

▶ Iterative Model:

❖ **Definition**

➢ The Iterative Model is a software development process where the system is developed in small sections called iterations. Each iteration involves planning, designing, implementing, and testing a part of the system, gradually refining the product.
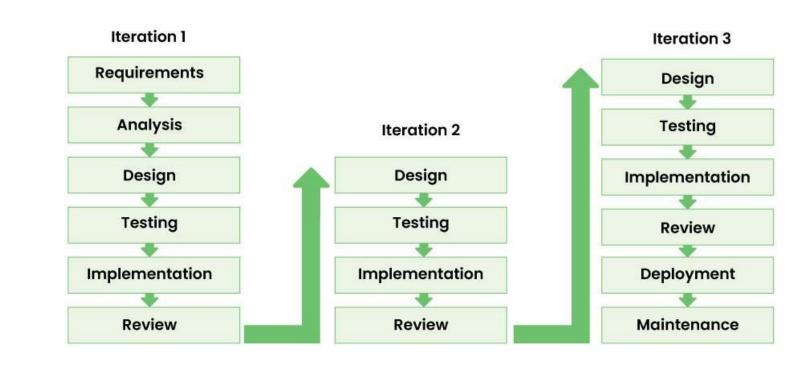
❖ **Characteristics**

➢ Development progresses through repeated cycles (iterations).

➢ Focus on refining functionality with each iteration.

➢ Allows feedback to shape subsequent iterations.

BCA 7th Sem - Shishir Ghimire

▶ Iterative Model:

❖ **Phases/Steps of the Model**

➤ **Requirement Gathering & Analysis:** Collect software requirements, analyze feasibility, and ensure the project stays within budget

➤ **Design:** Prepare the software design using diagrams such as Data Flow, Class, Activity, and State Transition diagrams.

➤ **Implementation:** Convert the design into code using programming languages; also called the coding phase.

➤ **Testing:** Test the software for bugs and errors using techniques like performance, security, stress, and requirement testing.

➤ **Deployment:** Deliver the software to the customer for use in their work environment.

➤ **Review:** Collect feedback and identify new requirements or errors. Repeat phases for improvements in the next iteration.

➤ **Maintenance:** Address feedback, fix issues, and update the software as needed.

Iterative Model:

Iteration 1
- Requirements
- Analysis
- Design
- Testing
- Implementation
- Review

Iteration 2
- Design
- Testing
- Implementation
- Review

Iteration 3
- Design
- Testing
- Implementation
- Review
- Deployment
- Maintenance

BCA 7th Sem - Shishir Ghimire

▶ Iterative Model:

❖ **Benefits of the Model**

➢ Risk Mitigation: Early detection and correction of issues.

➢ Flexibility: Adapts to changes in requirements.

➢ Customer Involvement: Frequent reviews ensure alignment with user needs.

➢ Gradual Development: Allows incremental improvement over time.

❖ **Drawbacks of the Model**

➢ Resource-Intensive: Requires more effort for planning and management.

➢ Incomplete Early Deliveries: Initial iterations may not be fully functional.

➢ Complex Management: Continuous feedback and iteration can complicate tracking.

▶ Iterative Model:

❖ **When to Use:**

➢ Large and complex projects with unclear or evolving requirements.

➢ Projects that require frequent testing and user feedback.

❖ **Example Cases:**

➢ Development of an AI-powered chatbot.

➢ A healthcare system with modules for patient records, billing, and reporting.

▶ **Spiral Model**

4.9.6

BCA 7th Sem - Shishir Ghimire

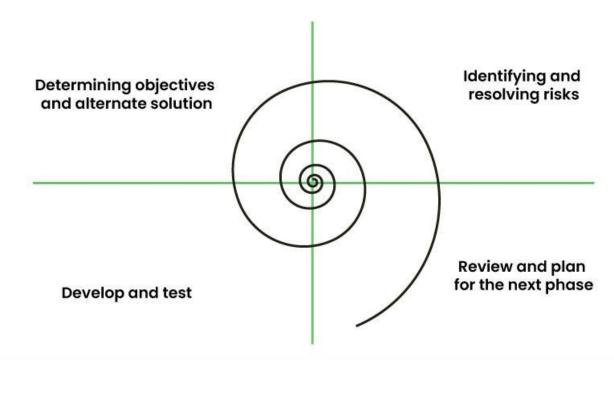# ▶ Spiral Model:

❖ **Definition**

➢ The Spiral Model is a risk-driven process model that combines iterative and waterfall approaches. It emphasizes risk assessment and management at each development stage.

❖ **Characteristics**

➢ Iterative with a strong focus on risk management.

➢ Combines elements of both iterative and sequential processes.

➢ Allows early identification of potential risks.

▶ Spiral Model:

❖ **Phases/Steps of the Model**

➢ **Determining Objectives and Alternate Solutions:**

■ Gather customer requirements, identify objectives, and propose alternative solutions.

➢ **Identifying and Resolving Risks:**

■ Evaluate solutions, identify risks, and resolve them using suitable strategies.

➢ **Develop and Test:**

■ Start software development by coding features and verifying them through testing.

➢ **Review and Plan for Next Phase:**

■ Deliver the current version to the customer for feedback and identify new requirements.

■ Plan for the next iteration (spiral).

▶ Spiral Model:

Determining objectives
and alternate solution

Identifying and
resolving risks

Develop and test

Review and plan
for the next phase

▶ Spiral Model:

❖ **Benefits of the Model**

➢ Risk Reduction: Regular risk assessments help avoid major setbacks.

➢ Flexibility: Adapts to changing requirements.

➢ Customer Feedback: Iterative development ensures user involvement.

➢ Scalability: Suitable for large and complex projects.

❖ **Drawbacks of the Model**

➢ Costly and Time-Consuming: Frequent risk evaluations add overhead.

➢ Requires Expertise: Demands experienced teams for effective risk management.

➢ Complexity: Managing spirals can be challenging in large projects.

▶ Spiral Model:

❖ **When to Use:**

➢ High-risk projects requiring robust risk management.

➢ Projects with evolving requirements and high complexity.

❖ **Example Cases:**

➢ Development of a national banking system.

➢ A military defense system with stringent safety and reliability standards.

Prototype Model   4.9.7
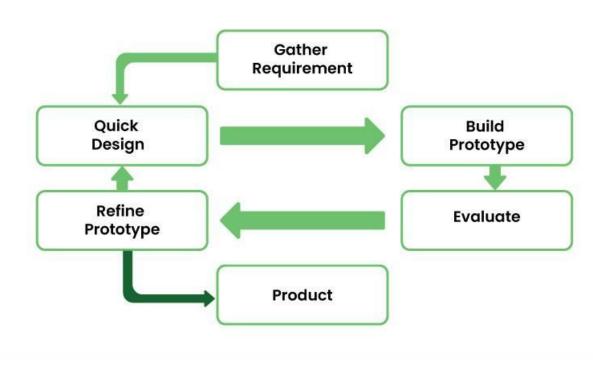
▶ Prototype Model:

❖ **Definition**

➢ The Prototype Model focuses on creating a working prototype of the software to demonstrate functionality and gather user feedback before full-scale development begins.

❖ **Characteristics**

➢ Early focus on user interface and functionality.

➢ Frequent interaction with users for feedback.

➢ Prototype may or may not evolve into the final product.

BCA 7th Sem - Shishir Ghimire

▶ Prototype Model:

❖ **Phases/Steps of the Model**

➢ **Requirement Gathering:** Collect initial requirements.

➢ **Quick Design:** Create a basic design for the prototype.

➢ **Prototype Development:** Develop a working prototype with key features.

➢ **User Evaluation:** Collect feedback from stakeholders.

➢ **Refinement:** Refine the prototype based on feedback until it meets user expectations.

➢ **Full-Scale Development:** Transition to full-scale development with a clear understanding of requirements.

BCA 7th Sem - Shishir Ghimire

▶ Prototype Model:

▶ Prototype Model:

❖ **Benefits of the Model**

➢ User Involvement: Ensures the product aligns with user needs.

➢ Requirement Clarity: Reduces misunderstandings about requirements.

➢ Early Problem Detection: Identifies design and functionality issues early.

❖ **Drawbacks of the Model**

➢ Time-Consuming: Frequent iterations can delay the actual development.

➢ Overreliance on Prototypes: Users may confuse the prototype with the final product.

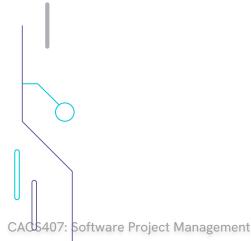➢ Costly: Building and refining prototypes can increase costs.

▶ Prototype Model:

❖ **When to Use:**

➢ Projects with unclear or ambiguous requirements.

➢ When user feedback is critical for success.

❖ **Example Cases:**

➢ Developing a mobile app with unique features requiring user validation.

➢ Designing an e-learning platform where usability is paramount.

▶ PYQs:

❖   Explain Cost Benefit Evaluation Techniques.

❖   What do you mean by Strategic Assessment? 5

❖   What do you mean by software evaluation? Explain different methods of software evaluation and costing. 10

THANKS!

Do you have any questions?

theciceerguy@p_m.me

9841893035

ghimires.com.np