

Software Engineering

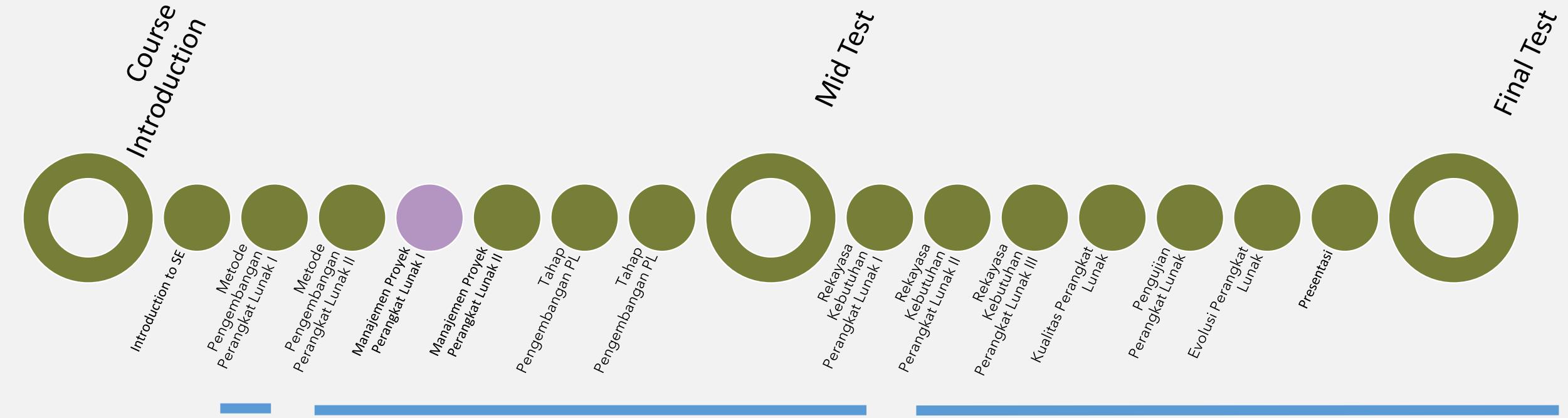
# Project Management

Program Studi Ilmu Komputer

Dosen: Intan Oktafiani, S.Kom, M.T.



# Weekly Materials



# Outline

Risk management

Managing people

Teamwork

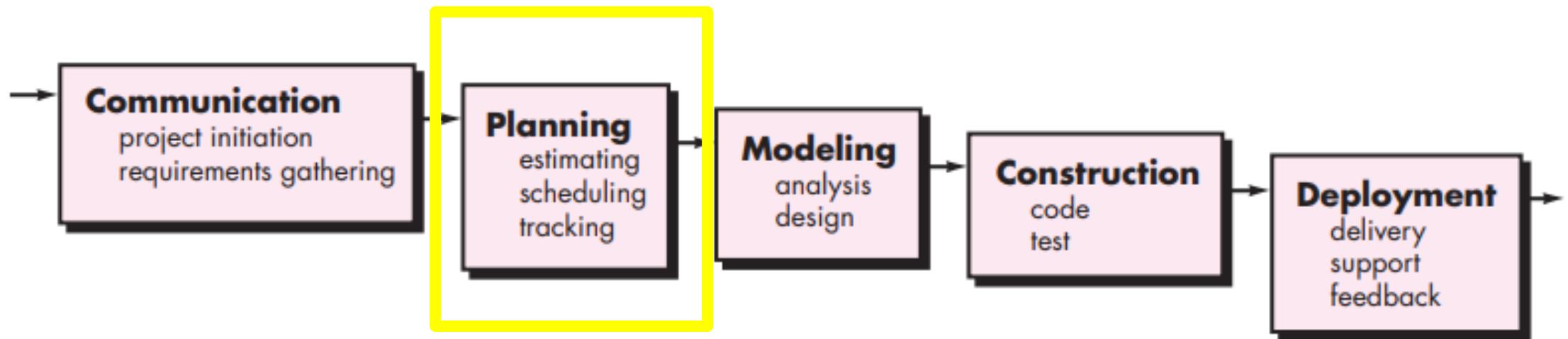
Software pricing

Project scheduling

Estimation techniques

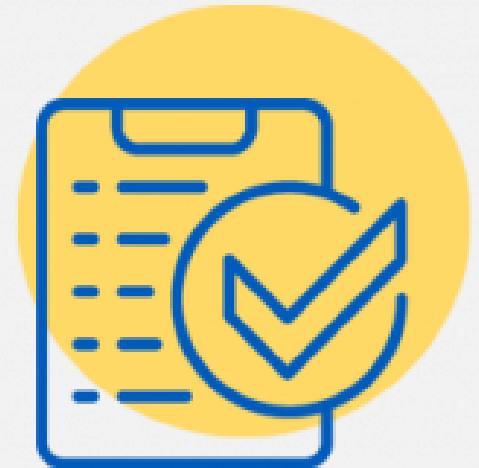
# The Waterfall Model

- Software Engineering - Pressman 7th



# Introduction

- *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**
- The project manager's job is to ensure that the software project **meets and overcomes these constraints** as well as **delivering high-quality software.**



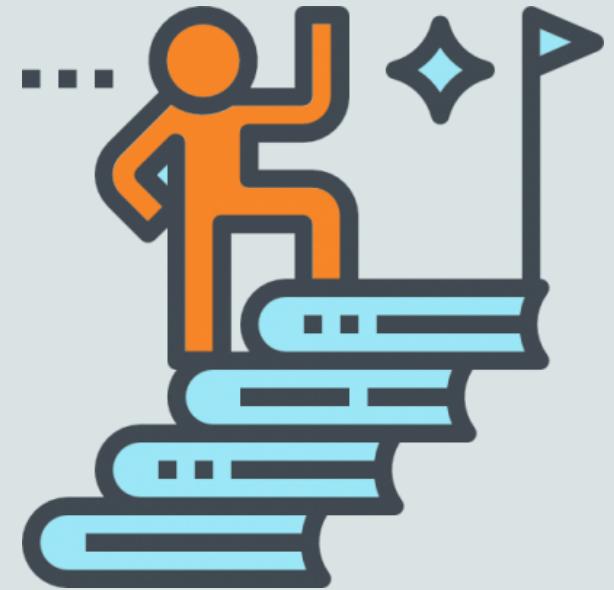
# Differents from other types of management

- Management of software projects is **different from other types of management** because:
  - The product is intangible
  - Large software projects are often 'one-off' projects
  - Software processes are variable and organization-specific

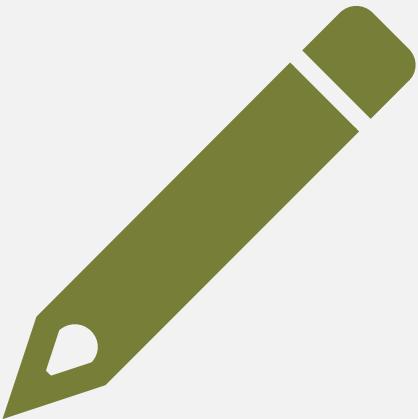


# The success criteria for project management

- Deliver the software to the customer at the agreed time.
- Keep overall costs within budget.
- Deliver software that meets the customer's expectations.
- Maintain a happy and well-functioning development team.



# Management activities



- Project planning
- Reporting
- Risk management
- People management
- Proposal writing

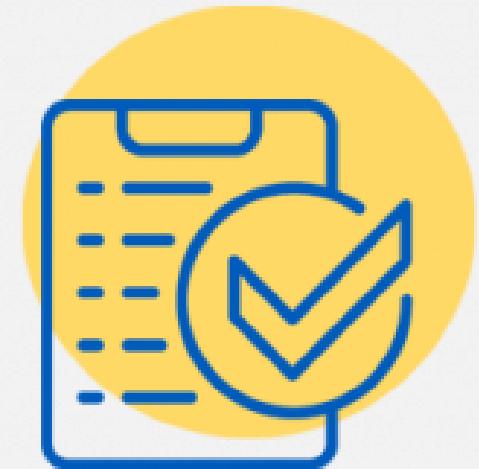
# Project planning

The project plan, which is created at the start of a project, is used to communicate how the work will be done to the project team and customers, and to help assess progress on the project.

# Introduction

**Rencana proyek dibuat pada awal proyek,**  
digunakan untuk mengkomunikasikan:

- bagaimana pekerjaan akan dilakukan kepada tim proyek dan pelanggan
- Untuk membantu menilai kemajuan proyek.



# Project planning life cycle



The proposal stage



The project startup phase

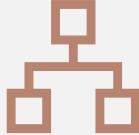


The project

# Project plans



Introduction



Work breakdown



Project organization



Project schedule



Risk analysis



Monitoring and reporting mechanisms



Hardware and software resource requirements

# Project plan supplements

Plan	Description
Quality plan	Describes the quality procedures and standards that will be used in a project.
Validation plan	Describes the approach, resources, and schedule used for system validation.
Configuration management plan	Describes the configuration management procedures and structures to be used.
Maintenance plan	Predicts the maintenance requirements, costs, and effort.
Staff development plan	Describes how the skills and experience of the project team members will be developed.

## Software pricing

The price of a software product to a customer is simply the cost of development plus profit for the developer. In practice, however, the relationship between the project cost and the price quoted to the customer is not usually so simple

# Main parameters to computing the costs of a software development project

- Effort costs (the costs of paying software engineers and managers)
- Hardware and software costs, including maintenance
- Travel and training costs.

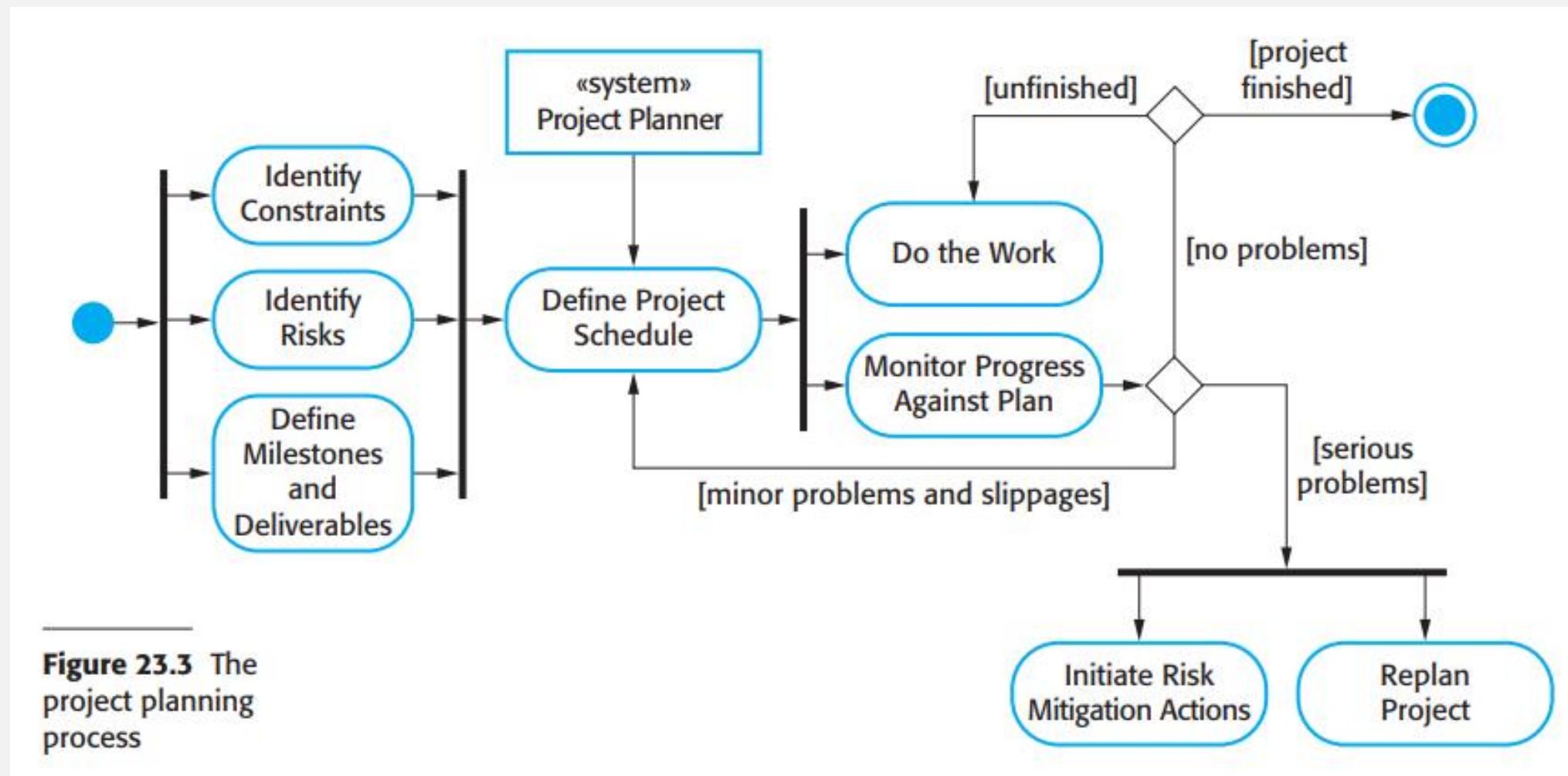
# Software pricing

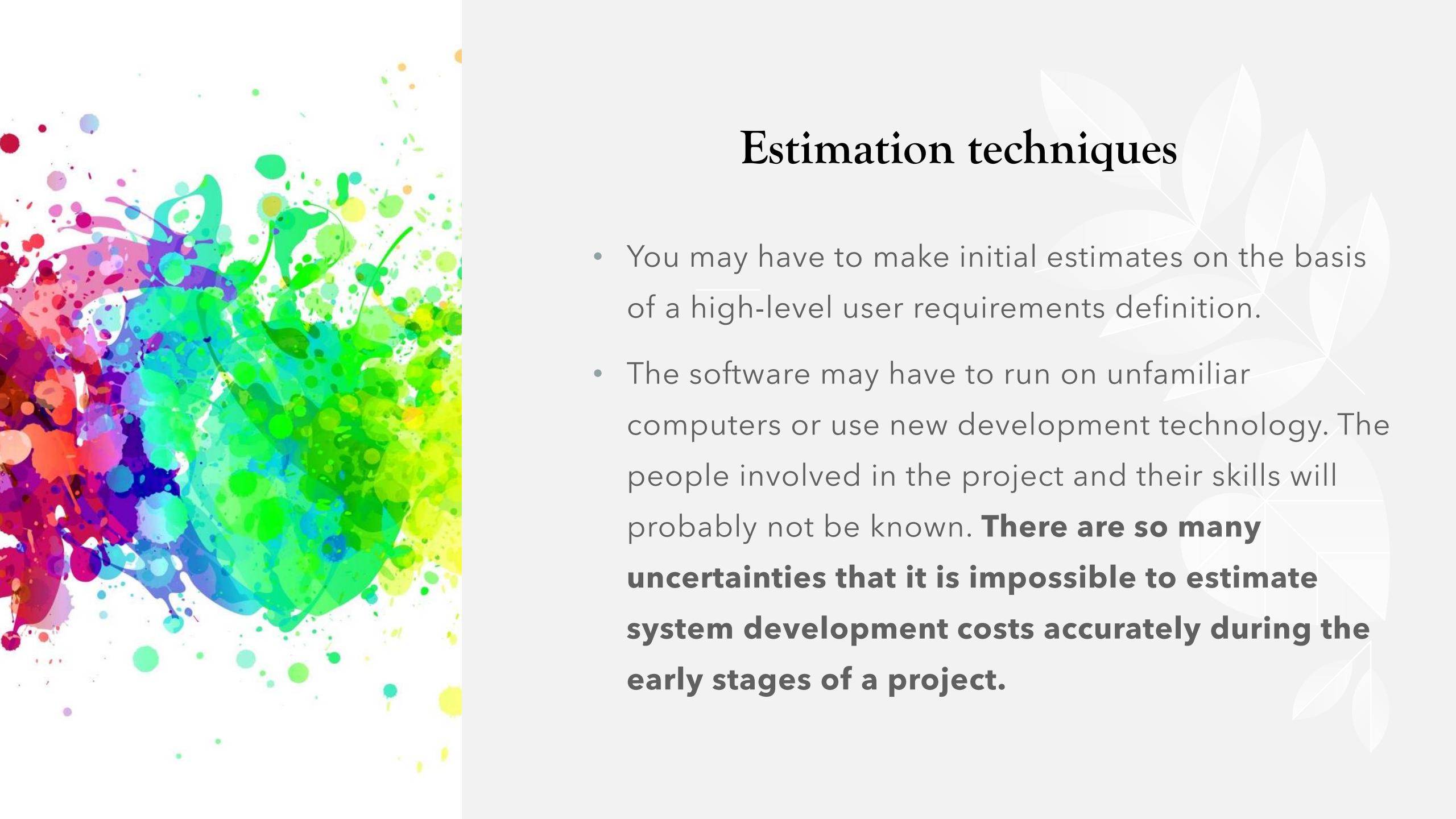
- You should take broader organizational, economic, political, and business considerations into account. You need to think about organizational concerns, the risks associated with the project, and the type of contract that will be used.
- **Deciding on a project price** should be a group activity involving marketing and sales staff, senior management, and project managers.

# Factors affecting software pricing

Factor	Description
Market opportunity	A development organization may quote a low price because it wishes to move into a new segment of the software market. Accepting a low profit on one project may give the organization the opportunity to make a greater profit later. The experience gained may also help it develop new products.
Cost estimate uncertainty	If an organization is unsure of its cost estimate, it may increase its price by a contingency over and above its normal profit.
Contractual terms	A customer may be willing to allow the developer to retain ownership of the source code and reuse it in other projects. The price charged may then be less than if the software source code is handed over to the customer.
Requirements volatility	If the requirements are likely to change, an organization may lower its price to win a contract. After the contract is awarded, high prices can be charged for changes to the requirements.
Financial health	Developers in financial difficulty may lower their price to gain a contract. It is better to make a smaller than normal profit or break even than to go out of business. Cash flow is more important than profit in difficult economic times.

# The planning process





# Estimation techniques

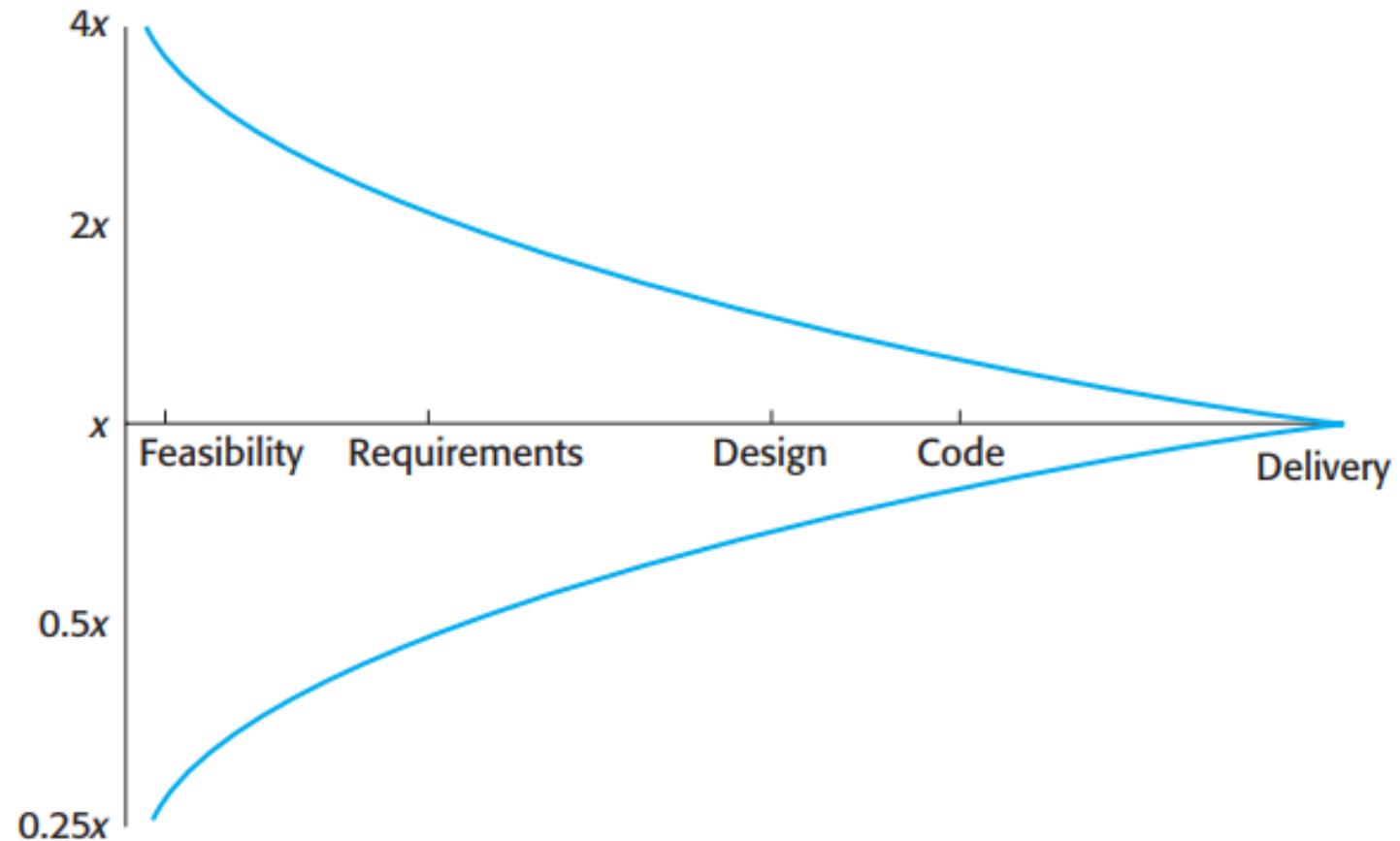
- You may have to make initial estimates on the basis of a high-level user requirements definition.
- The software may have to run on unfamiliar computers or use new development technology. The people involved in the project and their skills will probably not be known. **There are so many uncertainties that it is impossible to estimate system development costs accurately during the early stages of a project.**

# Techniques estimates effort and cost

- 1. Experience-based techniques** The estimate of future effort requirements is based on the manager's experience of past projects and the application domain. Essentially, the manager makes an informed judgment of what the effort requirements are likely to be.
- 2. Algorithmic cost modeling** In this approach, a formulaic approach is used to compute the project effort based on estimates of product attributes, such as size, and process characteristics, such as experience of staff involved.



# Estimate uncertainty





## Algorithmic cost modeling

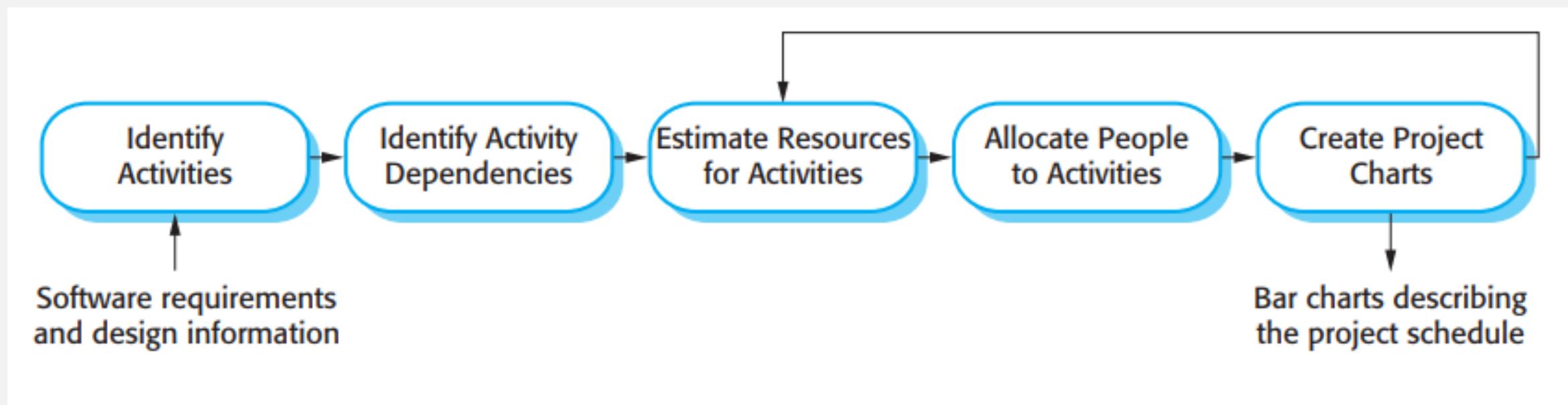
Algorithmic cost modeling uses a mathematical formula to predict project costs based on estimates of the project size; the type of software being developed; and other team, process, and product factors.

# **Project scheduling**

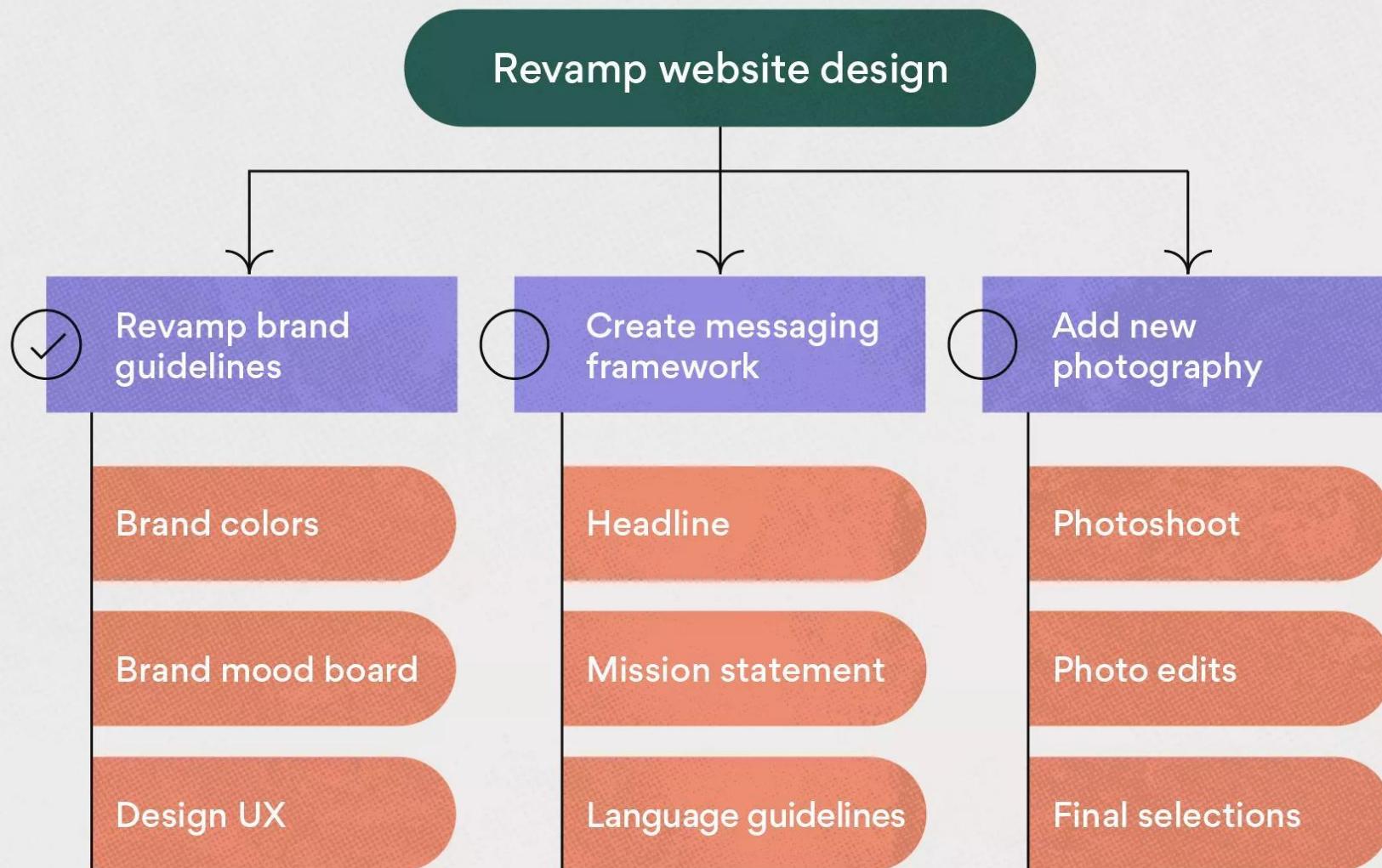
Project scheduling is the process of deciding how the work in a project will be organized as separate tasks, and when and how these tasks will be executed.

# The project scheduling process

Scheduling involves breaking down the total work involved in a project into separate tasks and estimating the time required to complete each task.



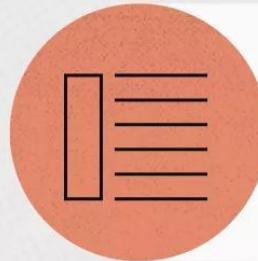
# Work breakdown structure example



# What's included in a work breakdown structure?



WBS dictionary



Task description



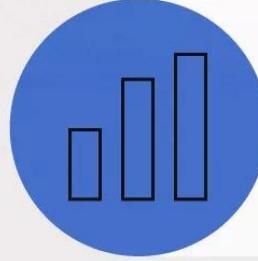
Task owner



Task budget



Completion date



Task status

# Project scheduling

- A good rule of thumb is to estimate as if nothing will go wrong, then increase your estimate to cover anticipated problems.
- A further contingency factor to cover unanticipated problems may also be added to the estimate. This extra contingency factor depends on the type of project, the process parameters (deadline, standards, etc.), and the quality and experience of the software engineers working on the project. **Contingency estimates may add 30% to 50% to the effort and time required for the project.**

# Two Types of Schedule Representation

1. **Bar charts**, which are calendar-based, show who is responsible for each activity, the expected elapsed time, and when the activity is scheduled to begin and end. Bar charts are sometimes called 'Gantt charts', after their inventor, Henry Gantt.
2. **Activity networks**, which are network diagrams, show the dependencies between the different activities making up a project.

# Project Management Tools

- Trello
- Microsoft Project
- Clickup
- Asana
- Jira



# Microsoft Project

Project

Email campaign to increase rider's aware... Oct 7 - Nov 1

Grid Board Timeline Zoom 6 Group members

Oct 6 Oct 20 Nov 3

1 Develop email campaign

2 Target audience profile

3 Create email storyboard

4 Marketing approval of storyboard

5 Define email response link points

6 Develop email message

7 Email campaign defined

8 Validate email messaging

9 Pilot email messaging inhouse

10 Refine email messaging

11 Refine email response link points

12 Final approval of email message

13 Email-ready

Add new task

Feedback

```
graph TD; T1[Develop email campaign] --> T2[Target audience profile]; T2 --> T3[Create email storyboard]; T3 --> T4[Marketing approval of storyboard]; T4 --> T5[Define email response link points]; T5 --> T6[Develop email message]; T6 --> T7[Email campaign defined]; T8[Validate email messaging] --> T9[Pilot email messaging inhouse]; T9 --> T10[Refine email messaging]; T10 --> T11[Refine email response link points]; T11 --> T12[Final approval of email message]; T12 --> T13[Email-ready]
```

# Asana Project Management

Events Go to any project or task...

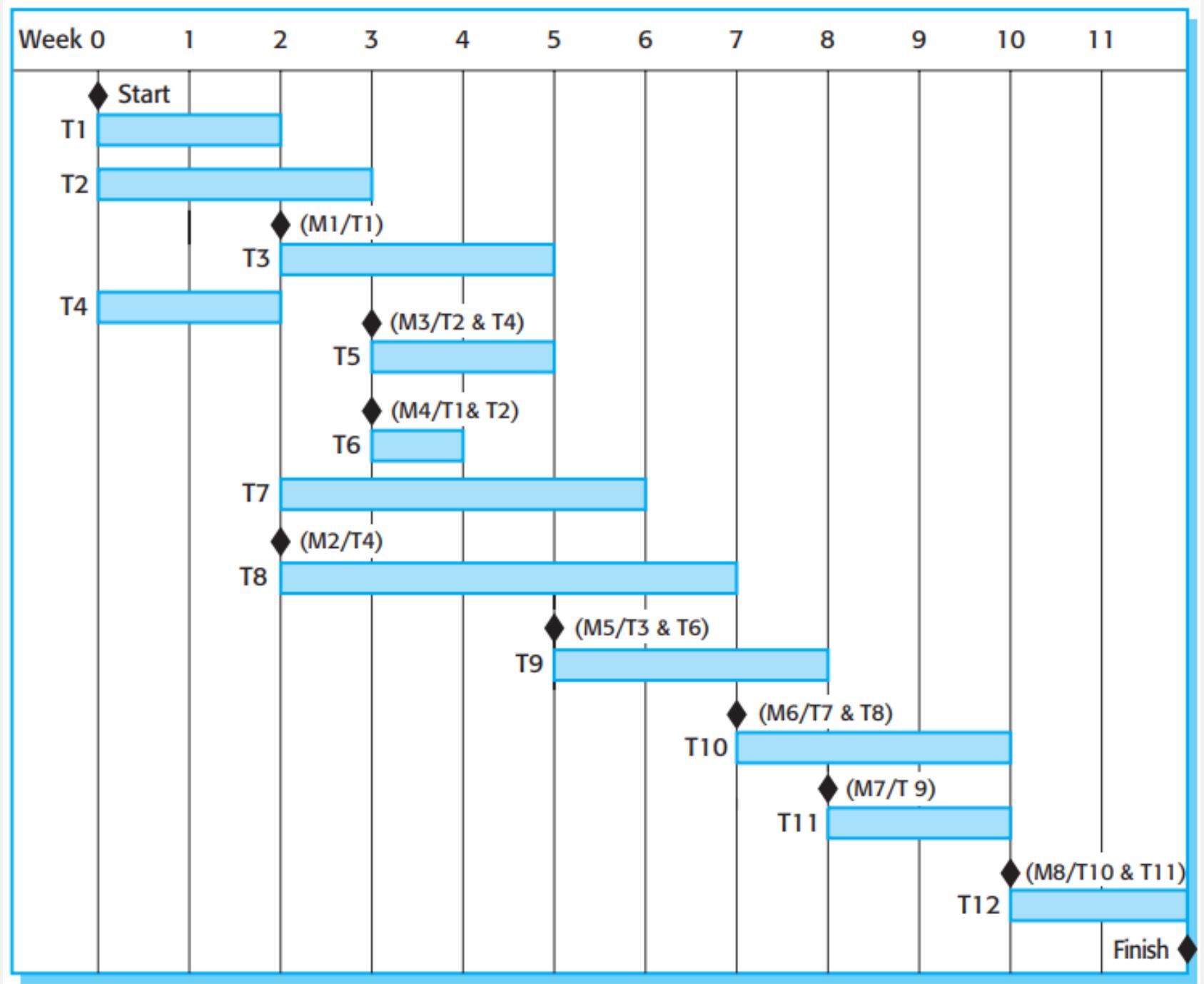
Add Project

	Status	Task Progress	Dates	Priority	Event type	
Seattle customer event Sales	● On Track 8 minutes ago	<div style="width: 76%;">76%</div>	Jan 14 – Feb 19	High	Full d...	
Mexico City customer event Sales	● At Risk 8 minutes ago	<div style="width: 85%;">85%</div>	Dec 30, 2018 – Feb 15, 2019	Medi...	Dinner	
Paris customer event Sales · 3 overdue tasks	● On Track 7 minutes ago	<div style="width: 23%;">23%</div>	Mar 18 – Apr 30	High	Dinner	
Buenos Aires customer event Sales	● On Track 7 minutes ago	<div style="width: 43%;">43%</div>	Feb 4 – Mar 15	Low	Work...	
Sydney customer event Sales · 5 overdue tasks	● Off Track 7 minutes ago	<div style="width: 19%;">19%</div>	Feb 8 – Apr 30	Low	Full d...	
New York customer event Sales	● At Risk 7 minutes ago	<div style="width: 23%;">23%</div>	Apr 1 – 30	Low	Dinner	
London customer event Marketing	● On Track 7 minutes ago	<div style="width: 80%;">80%</div>	Jan 28 – May 17	High	Full d...	

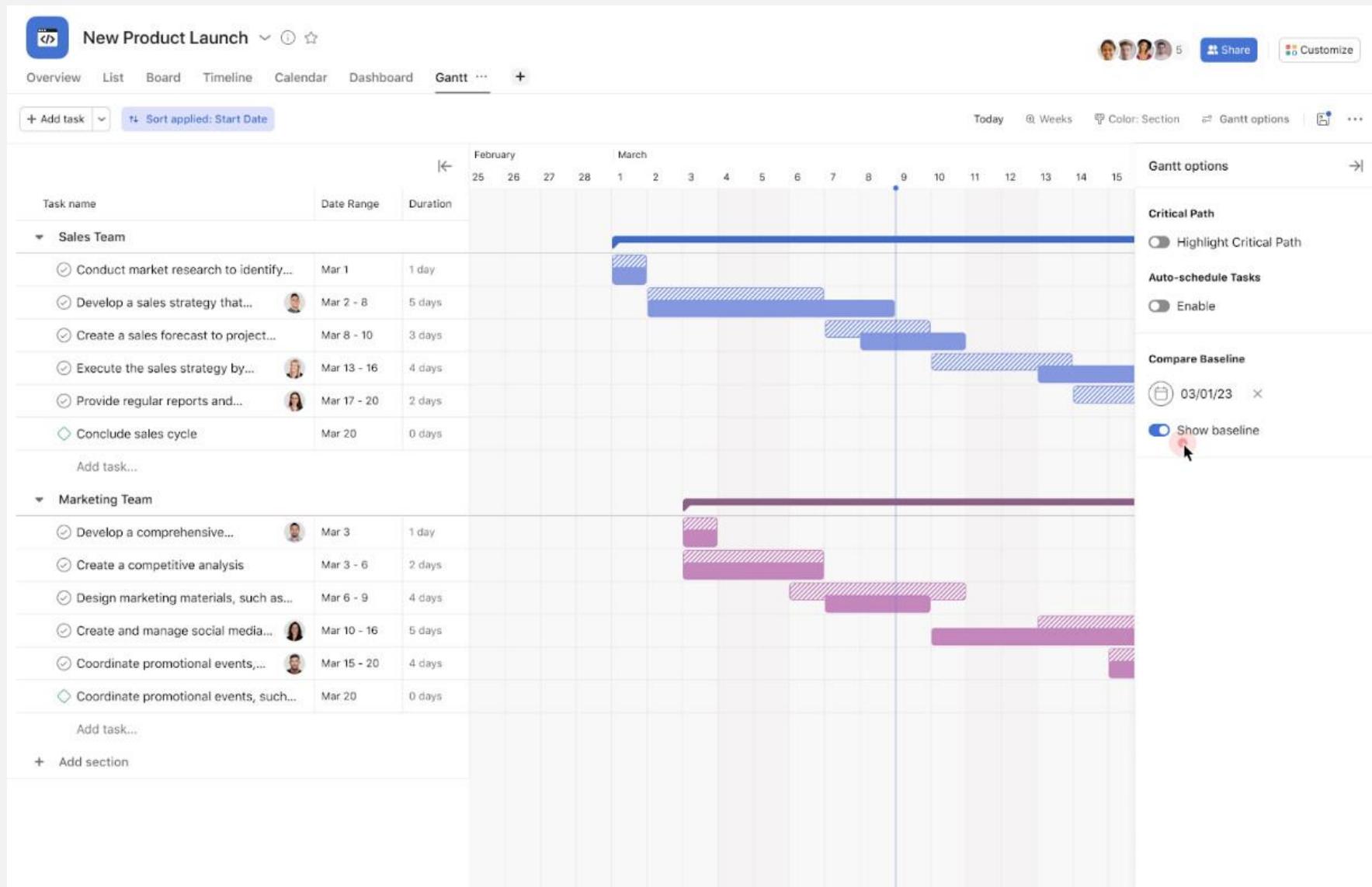
# Project activities

1. A duration in calendar days or months.
2. An effort estimate, which reflects the number of person-days or person-months to complete the work.
3. A deadline by which the activity should be completed.
4. A defined endpoint. This represents the tangible result of completing the activity. This could be a document, the holding of a review meeting, the successful execution of all tests, etc.

# Activity bar chart/ Gantt chart



# Asana Gantt Chart



# Tasks, durations, and dependencies

Task	Effort (person-days)	Duration (days)	Dependencies
T1	15	10	
T2	8	15	
T3	20	15	T1 (M1)
T4	5	10	
T5	5	10	T2, T4 (M3)
T6	10	5	T1, T2 (M4)
T7	25	20	T1 (M1)
T8	75	25	T4 (M2)
T9	10	15	T3, T6 (M5)
T10	20	15	T7, T8 (M6)
T11	10	10	T9 (M7)
T12	20	10	T10, T11 (M8)

# Staff allocation chart

A Gantt chart illustrating task assignments across 11 weeks for seven team members: Jane, Ali, Geetha, Maya, Fred, Mary, and Hong. The x-axis represents weeks from 0 to 11. Each team member's tasks are represented by blue bars.

- Jane:** T1 (Week 0-1), T3 (Week 2-5), T9 (Week 5-7), T10 (Week 7-9), T12 (Week 10-11).
- Ali:** T1 (Week 0-1), T8 (Week 2-7).
- Geetha:** T2 (Week 0-2), T6 (Week 3-4), T7 (Week 4-5), T10 (Week 7-10).
- Maya:** T3 (Week 3-4), T8 (Week 4-7).
- Fred:** T4 (Week 0-2), T8 (Week 2-7), T11 (Week 7-11), T12 (Week 11-11).
- Mary:** T5 (Week 3-5).
- Hong:** T7 (Week 2-6), T6 (Week 4-4).

# Risk Management

**Risk** = some adverse circumstance that may happen and affect negatively the project, the product, and/or the business

**Risk management** means anticipating risks and preparing plans to reduce their effect



## Categories of risk

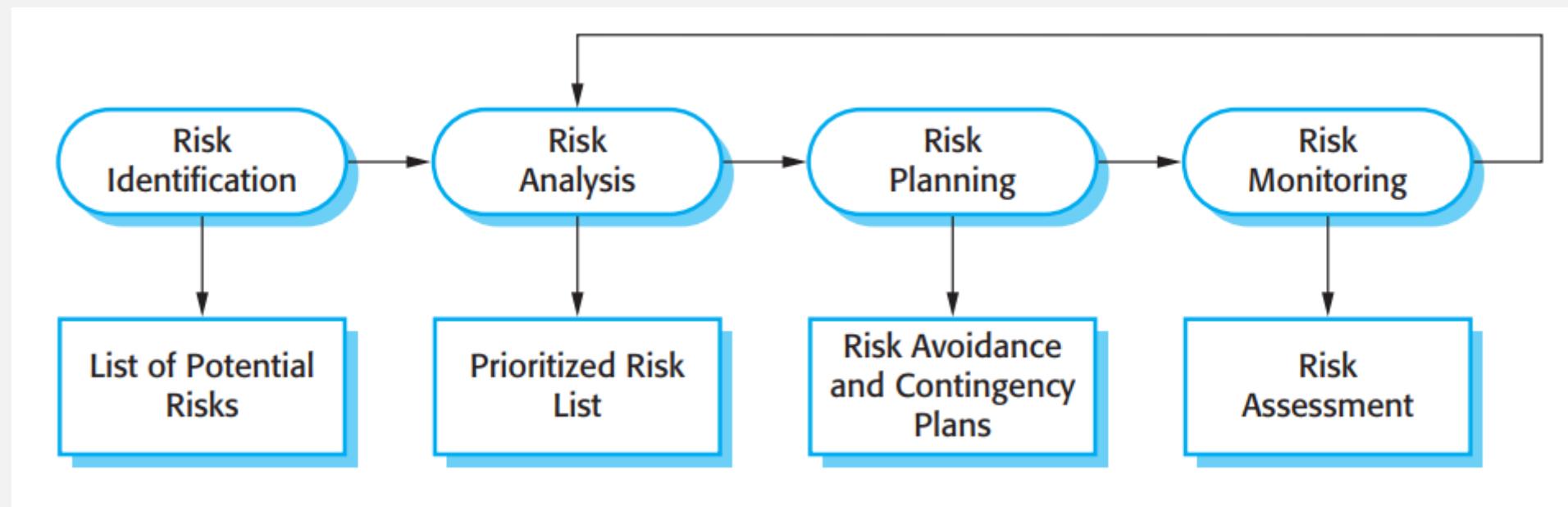
---

- Project risks
- Product risks
- Business risks

# Examples of common project, product, and business risks

Risk	Affects	Description
Staff turnover	Project	Experienced staff will leave the project before it is finished.
Management change	Project	There will be a change of organizational management with different priorities.
Hardware unavailability	Project	Hardware that is essential for the project will not be delivered on schedule.
Requirements change	Project and product	There will be a larger number of changes to the requirements than anticipated.
Specification delays	Project and product	Specifications of essential interfaces are not available on schedule.
Size underestimate	Project and product	The size of the system has been underestimated.
CASE tool underperformance	Product	CASE tools, which support the project, do not perform as anticipated.
Technology change	Business	The underlying technology on which the system is built is superseded by new technology.
Product competition	Business	A competitive product is marketed before the system is completed.

# The risk management process



# Risk identification

- Concerned with identifying the risks that could pose a major threat to the software engineering process, the software being developed, or the development organization.
- Risk identification may be a team process where a team get together to brainstorm possible risks.

# Type of Risk

Risk type	Possible risks
Technology	The database used in the system cannot process as many transactions per second as expected. (1) Reusable software components contain defects that mean they cannot be reused as planned. (2)
People	It is impossible to recruit staff with the skills required. (3) Key staff are ill and unavailable at critical times. (4) Required training for staff is not available. (5)
Organizational	The organization is restructured so that different management are responsible for the project. (6) Organizational financial problems force reductions in the project budget. (7)
Tools	The code generated by software code generation tools is inefficient. (8) Software tools cannot work together in an integrated way. (9)
Requirements	Changes to requirements that require major design rework are proposed. (10) Customers fail to understand the impact of requirements changes. (11)
Estimation	The time required to develop the software is underestimated. (12) The rate of defect repair is underestimated. (13) The size of the software is underestimated. (14)

# Risk analysis

## ***Estimate risk probability:***

- Very low (< 10%)
- Low (10-25%)
- Moderate (25-50%)
- High (50-75%)
- Very high (> 75%)

## ***Establish risk seriousness:***

- Insignificant
- Tolerable
- Serious
- Catastrophic

# Risk analysis

Risk	Probability	Effects
Organizational financial problems force reductions in the project budget (7).	Low	Catastrophic
It is impossible to recruit staff with the skills required for the project (3).	High	Catastrophic
Key staff are ill at critical times in the project (4).	Moderate	Serious
Faults in reusable software components have to be repaired before these components are reused. (2).	Moderate	Serious
Changes to requirements that require major design rework are proposed (10).	Moderate	Serious
The organization is restructured so that different management are responsible for the project (6).	High	Serious
The database used in the system cannot process as many transactions per second as expected (1).	Moderate	Serious
The time required to develop the software is underestimated (12).	High	Serious
Software tools cannot be integrated (9).	High	Tolerable
Customers fail to understand the impact of requirements changes (11).	Moderate	Tolerable
Required training for staff is not available (5).	Moderate	Tolerable
The rate of defect repair is underestimated (13).	Moderate	Tolerable
The size of the software is underestimated (14).	High	Tolerable
Code generated by code generation tools is inefficient (8).	Moderate	Insignificant

# Risk planning

- Risk planning means preparing a strategy to deal with each of the risks identified
- Classes of strategies:
  - Avoidance strategies: the probability of the risk will be diminished
  - Minimization strategies: the effect of the risk will be reduced
  - Contingency strategies: plans for the worst case scenarios

# Risk planning

Risk	Strategy
Organizational financial problems	Prepare a briefing document for senior management showing how the project is making a very important contribution to the goals of the business and presenting reasons why cuts to the project budget would not be cost-effective.
Recruitment problems	Alert customer to potential difficulties and the possibility of delays; investigate buying-in components.
Staff illness	Reorganize team so that there is more overlap of work and people therefore understand each other's jobs.
Defective components	Replace potentially defective components with bought-in components of known reliability.
Requirements changes	Derive traceability information to assess requirements change impact; maximize information hiding in the design.
Organizational restructuring	Prepare a briefing document for senior management showing how the project is making a very important contribution to the goals of the business.
Database performance	Investigate the possibility of buying a higher-performance database.
Underestimated development time	Investigate buying-in components; investigate use of a program generator.

# Risk monitoring

- Risk monitoring is the process of checking that your assumptions about the product, process, and business risks have not changed.
- You should regularly assess each of the identified risks to decide whether or not that risk is becoming more or less probable.
- You should also think about whether or not the effects of the risk have changed.
- You should monitor risks regularly at all stages in a project. At every management review, you should consider and discuss each of the key risks separately.
- You should decide if the risk is more or less likely to arise and if the seriousness and consequences of the risk have changed.

# Risk indicators

Risk type	Potential indicators
Technology	Late delivery of hardware or support software; many reported technology problems.
People	Poor staff morale; poor relationships amongst team members; high staff turnover.
Organizational	Organizational gossip; lack of action by senior management.
Tools	Reluctance by team members to use tools; complaints about CASE tools; demands for higher-powered workstations.
Requirements	Many requirements change requests; customer complaints.
Estimation	Failure to meet agreed schedule; failure to clear reported defects.

# **People Management**

The people working in a software organization are its greatest assets

# 4 critical factors in people management

- Consistency
- Respect
- Inclusion
- Honesty



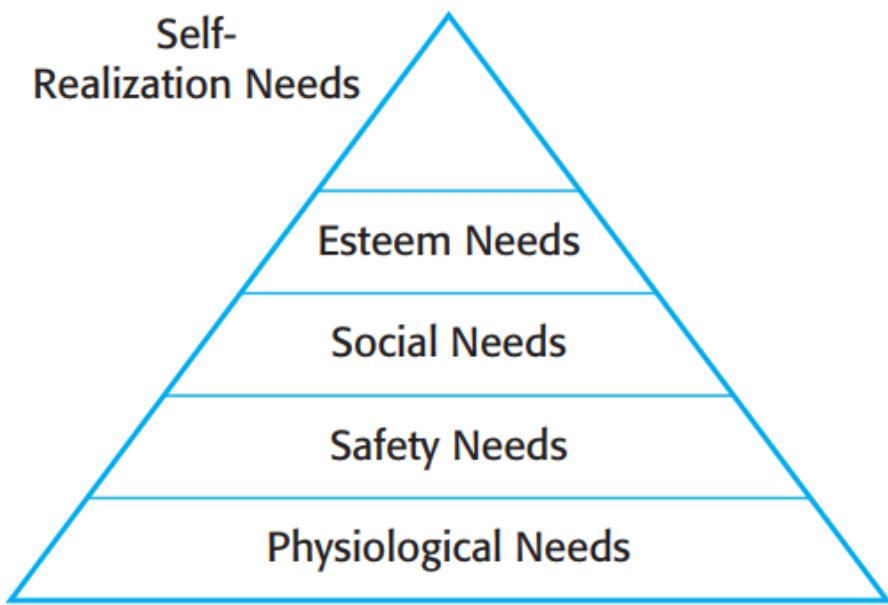
# Motivating people

- We need to motivate the people that work with, so that they contribute to **the best of their abilities**.
  - Motivation means **organizing the work and the working environment to encourage people to work as effectively as possible**.
- If people are not motivated, they will not be interested in the work they are doing. They will work slowly, be more likely to make mistakes, and will not contribute to the broader goals of the team or the organization.



# Motivating people

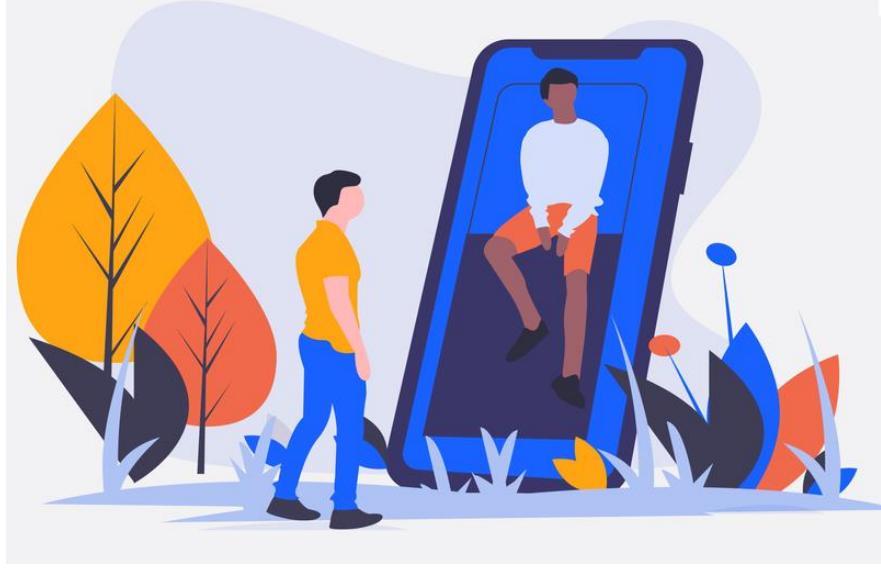
---



Maslow (1954)

Maslow suggests that people are motivated by satisfying their needs.

- The lower levels of this hierarchy represent the fundamental needs for food, sleep, and so on, and the need to feel secure in an environment.
- **Social needs** are concerned with the need to feel part of a social grouping.
- **Esteem needs** represent the need to feel respected by others, and self-realization needs are concerned with personal development.



## Social needs

---

- Give people time to meet their co-workers and provide places for them to meet.

## Esteem needs

---



- show people that they are valued by the organization

## Self-realization needs

---



- give people responsibility for their work, assign them demanding tasks, and provide a training programme where people can develop their skills.

# Teamwork

Most professional software is developed by project teams that range in size from two to several hundred people.

# Teamwork

- Selecting group members
- Group organization
- Group communications

# The benefits of creating a cohesive group

- The group can establish its own quality standards
- Individuals learn from and support each other
- Knowledge is shared
- Refactoring and continual improvement is encouraged

# Teamwork

- *The people in the group* You need a mix of people in a project group as software development involves diverse activities such as negotiating with clients, programming, testing, and documentation.
- *The group organization* A group should be organized so that individuals can contribute to the best of their abilities and tasks can be completed as expected.
- *Technical and managerial communications* Good communications between group members, and between the software engineering team and other project stakeholders, is essential



Thank you