



M.Sc. in Computer Science  
Department of Computer Science  
Faculty of Applied Sciences  
University of Sri Jayewardenepura

CSC 540 2.0    Software Engineering

**Presented By:**

**Surani Tissera(PhD)**

**Department of Computer Science**

# Project Management

2



This is a property of Department of Computer Science, Faculty of Applied Science, University of Sri Jayewardenepura.

# Objectives

You will:

- know the principal tasks of software project managers;
- have been introduced to the notion of risk management and some of the risks that can arise in software projects;
- understand factors that influence personal motivation and what these might mean for software project managers;
- understand key issues that influence team working, such as team composition, organization, and communication.



# Topic Covered

- Risk management
- Managing people
- Project scheduling

- 



# Software project management

- Concerned with activities involved in ensuring that software is delivered on time and on schedule and in accordance with the requirements of the organisations developing and procuring the software.
- Project management is needed because software development is always subject to budget and schedule constraints that are set by the organisation developing the software.



# Success criteria

- 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
  - For planning.
  - Estimating and scheduling project development
  - Assigning people to tasks.
- Reporting
  - Progress of a project to customers
  - Progress of a project to the managers of the company
- Risk management
  - Assess the risks that may affect a project,
  - Monitor these risks

Take action when problems arise.



# Management activities

- People management
  - Have to choose people for their team
  - Establish ways of working that leads to effective team performance
- Proposal writing
  - Writing a proposal to win a contract to carry out an item of work.
  - The proposal describes the objectives of the project and how it will be carried out.





# Risk management

- Risk management is concerned with identifying risks and drawing up plans to minimise their effect on a project.
- A risk is a probability that some adverse circumstance will occur
  - Project risks affect schedule or resources;
  - Product risks affect the quality or performance of the software being developed;
  - Business risks affect the organisation developing or procuring the software.

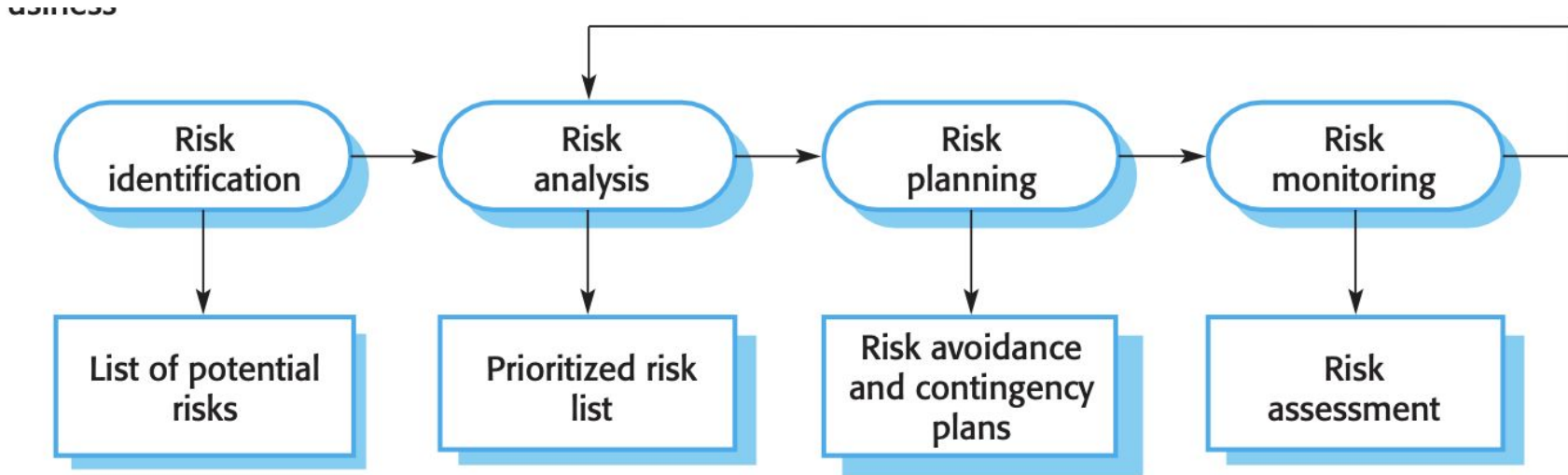


# Examples of common project, product, and business risks

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 company 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.
Software tool underperformance	Product	Software tools that 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.



# Risk management Process



# Risk Identification

- May be a team activities or based on the individual project manager's experience.
- A checklist of common risks may be used to identify risks in a project
  - Technology risks.
  - People risks.
  - Organisational risks.
  - Requirements risks.
  - Estimation risks.



# Examples of different risk types

Risk type	Possible risks
Estimation	<ol style="list-style-type: none"><li>1. The time required to develop the software is underestimated.</li><li>2. The rate of defect repair is underestimated.</li><li>3. The size of the software is underestimated.</li></ol>
Organizational	<ol style="list-style-type: none"><li>4. The organization is restructured so that different management are responsible for the project.</li><li>5. Organizational financial problems force reductions in the project budget.</li></ol>
People	<ol style="list-style-type: none"><li>6. It is impossible to recruit staff with the skills required.</li><li>7. Key staff are ill and unavailable at critical times.</li><li>8. Required training for staff is not available.</li></ol>
Requirements	<ol style="list-style-type: none"><li>9. Changes to requirements that require major design rework are proposed.</li><li>10. Customers fail to understand the impact of requirements changes.</li></ol>
Technology	<ol style="list-style-type: none"><li>11. The database used in the system cannot process as many transactions per second as expected.</li><li>12. Faults in reusable software components have to be repaired before these components are reused.</li></ol>
Tools	<ol style="list-style-type: none"><li>13. The code generated by software code generation tools is inefficient.</li><li>14. Software tools cannot work together in an integrated way.</li></ol>



# Risk Analysis

- Assess probability and seriousness of each risk.
- Probability may be very low, low, moderate, high or very high.
- Risk consequences might be catastrophic, serious, tolerable or insignificant.



# Risk types and examples

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



# Risk planning

- Consider each risk and develop a strategy to manage that risk.
- Avoidance strategies
  - The probability that the risk will arise is reduced;
- Minimisation strategies
  - The impact of the risk on the project or product will be reduced;
- Contingency plans
  - If the risk arises, contingency plans are plans to deal with that risk;





# Strategies to help manage risk

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 automated code generation.



# Risk Monitoring

- Assess each identified risks regularly to decide whether or not it is becoming less or more probable.
- Also assess whether the effects of the risk have changed.
- Each key risk should be discussed at management progress meetings.



# Risk indicators

Risk type	Potential indicators
Estimation	Failure to meet agreed schedule; failure to clear reported defects.
Organizational	Organizational gossip; lack of action by senior management.
People	Poor staff morale; poor relationships among team members; high staff turnover.
Requirements	Many requirements change requests; customer complaints.
Technology	Late delivery of hardware or support software; many reported technology problems.
Tools	Reluctance by team members to use tools; complaints about software tools; requests for faster computers/more memory, and so on.



# Managing People

- People are an organisation's most important assets.
- The tasks of a manager are essentially people-oriented. Unless there is some understanding of people, management will be unsuccessful.
- Poor people management is an important contributor to project failure.



# Managing People

- Consistency
  - Team members should all be treated in a comparable way without favourites or discrimination.
- Respect
  - Different team members have different skills and these differences should be respected.
- Inclusion
  - Involve all team members and make sure that people's views are considered.
- Honesty
  - You should always be honest about what is going well and what is going badly in a project.



# Motivating people

- An important role of a manager is to motivate the people working on a project.
- Motivation means organizing the work and the working environment to encourage people to work effectively.
  - 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

- Motivation is a complex issue but it appears that there are different types of motivation based on:
  - Basic needs (e.g. food, sleep, etc.);
  - Personal needs (e.g. respect, self-esteem);
  - Social needs (e.g. to be accepted as part of a group).



# Team Work

- Most software engineering is a group activity
  - The development schedule for most non-trivial software projects is such that they cannot be completed by one person working alone.
- A good group is cohesive and has a team spirit. The people involved are motivated by the success of the group as well as by their own personal goals.
- Group interaction is a key determinant of group performance.
- Flexibility in group composition is limited
  - Managers must do the best they can with available people.





# Group cohesiveness

- In a cohesive group, members consider the group to be more important than any individual in it.
- The advantages of a cohesive group are:
  - Group quality standards can be developed by the group members.
  - Team members learn from each other and get to know each other's work; Inhibitions caused by ignorance are reduced.
  - Knowledge is shared. Continuity can be maintained if a group member leaves.
  - Refactoring and continual improvement is encouraged. Group members work collectively to deliver high quality results and fix problems, irrespective of the individuals who originally created the design or program.



# Selecting group members

- A manager or team leader's job is to create a cohesive group and organize their group so that they can work together effectively.
- This involves creating a group with the right balance of technical skills and personalities, and organizing that group so that the members work together effectively.



# Assembling a Team

- ✧ May not be possible to appoint the ideal people to work on a project
  - Project budget may not allow for the use of highly-paid staff;
  - Staff with the appropriate experience may not be available;
  - An organisation may wish to develop employee skills on a software project.
- ✧ Managers have to work within these constraints especially when there are shortages of trained staff.



# Group Composition

- ✧ Group composed of members who share the same motivation can be problematic
  - Task-oriented - everyone wants to do their own thing;
  - Self-oriented - everyone wants to be the boss;
  - Interaction-oriented - too much chatting, not enough work.
- ✧ An effective group has a balance of all types.
- ✧ This can be difficult to achieve software engineers are often task-oriented.
- ✧ Interaction-oriented people are very important as they can



ect and defuse tensions that arise.

# Project Planning

- Project planning involves breaking down the work into parts and assign these to project team members, anticipate problems that might arise and prepare tentative solutions to those problems.
- 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.



# Planning Stages

- **At the proposal stage**, when you are bidding for a contract to develop or provide a software system.
- **During the project startup phase**, when you have to plan who will work on the project, how the project will be broken down into increments, how resources will be allocated across your company, etc.
- **Periodically throughout the project**, when you modify your plan in the light of experience gained and information from monitoring the progress of the work.



# Proposal Planning

- Planning may be necessary with only **outline software requirements**.
- The aim of planning at this stage is to **provide information that will be used in setting a price for the system to customers**.



# Software Pricing

- Estimates are made to discover the cost, to the developer, of producing a software system.
  - You take into account, hardware, software, travel, training and effort costs.
- There is **not** a simple relationship between the development cost and the price charged to the customer.
- Broader organisational, economic, political and business considerations influence the price charged.





# Factors affecting software pricing

Factor	Description
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 might then be reduced to reflect the value of the source code to the developer.
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.
Financial health	Companies with financial problems 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.
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.
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.



# Project Plan

In a plan-driven development project, a project plan sets out the resources available to the project, the work breakdown and a schedule for carrying out the work.

## Plan sections

- Introduction
- Project organization
- Risk analysis
- Hardware and software resource requirements
- Work breakdown
- Project schedule



## Monitoring and reporting mechanisms

This is a property of Department of Computer Science, Faculty of Applied Science, University of Sri Jayewardenepura.

# Project plan supplements

Plan	Description
Configuration management plan	Describes the configuration management procedures and structures to be used.
Deployment plan	Describes how the software and associated hardware (if required) will be deployed in the customer's environment. This should include a plan for migrating data from existing systems.
Maintenance plan	Predicts the maintenance requirements, costs, and effort.
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.

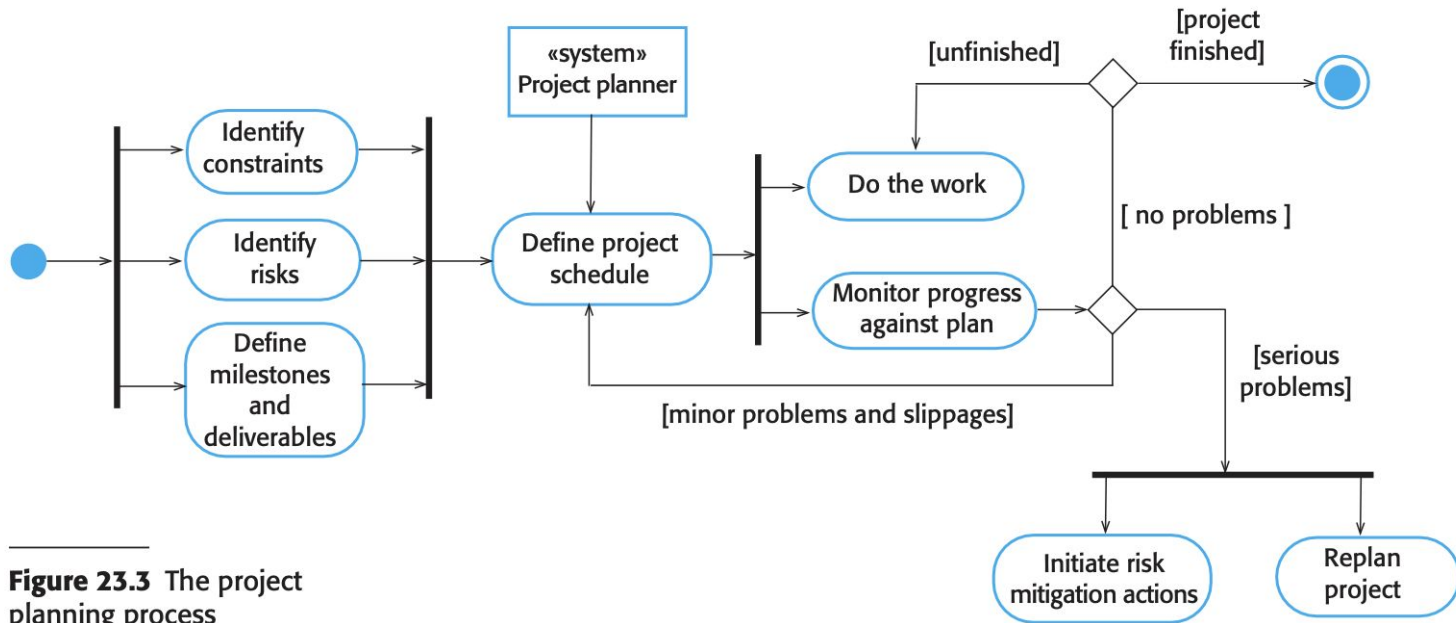


# The planning process

- ✧ Project planning is an iterative process that starts when you create an initial project plan during the project startup phase.
- ✧ Plan changes are inevitable.
  - As more information about the system and the project team becomes available during the project, you should regularly revise the plan to reflect requirements, schedule and risk changes.
  - Changing business goals also leads to changes in project plans. As business goals change, this could affect all projects, which may then have to be re-planned.



# The planning process



**Figure 23.3** The project planning process



# 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.
- You estimate the calendar time needed to complete each task, the effort required and who will work on the tasks that have been identified.
- You also have to estimate the resources needed to complete each task, such as the disk space required on a server, the time required on specialized hardware, such as a simulator, and what the travel budget will be.



# Project scheduling activities

- Split project into tasks and estimate time and resources required to complete each task.
- Organize tasks concurrently to make optimal use of workforce.
- Minimize task dependencies to avoid delays caused by one task waiting for another to complete.
- Dependent on project managers intuition and experience.



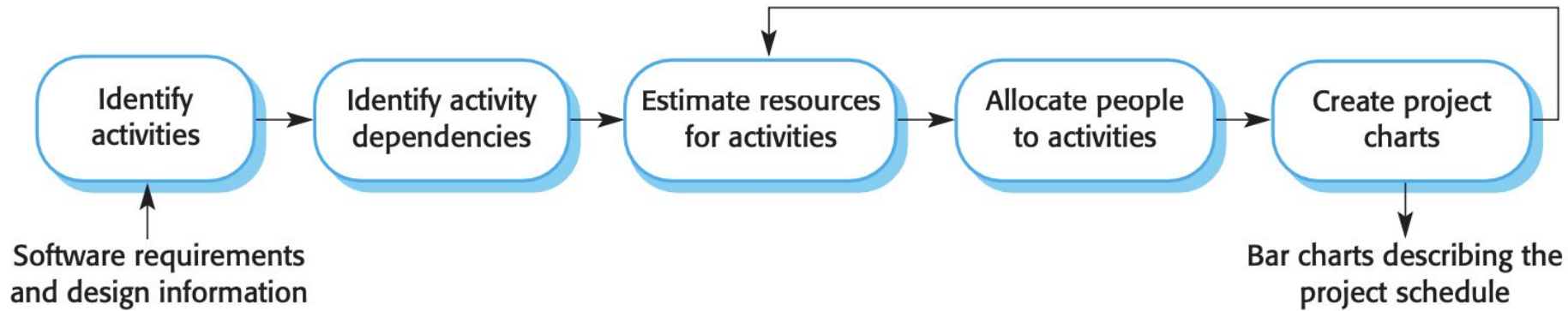
# Milestones and deliverables

- Milestones are points in the schedule against which you can assess progress, for example, the handover of the system for testing.
- Deliverables are work products that are delivered to the customer, e.g. a requirements document for the system.





# The project scheduling process



# Scheduling problems

- Estimating the difficulty of problems and hence the cost of developing a solution is hard.
- Productivity is not proportional to the number of people working on a task.
- Adding people to a late project makes it later because of communication overheads.
- The unexpected always happens. Always allow contingency in planning.

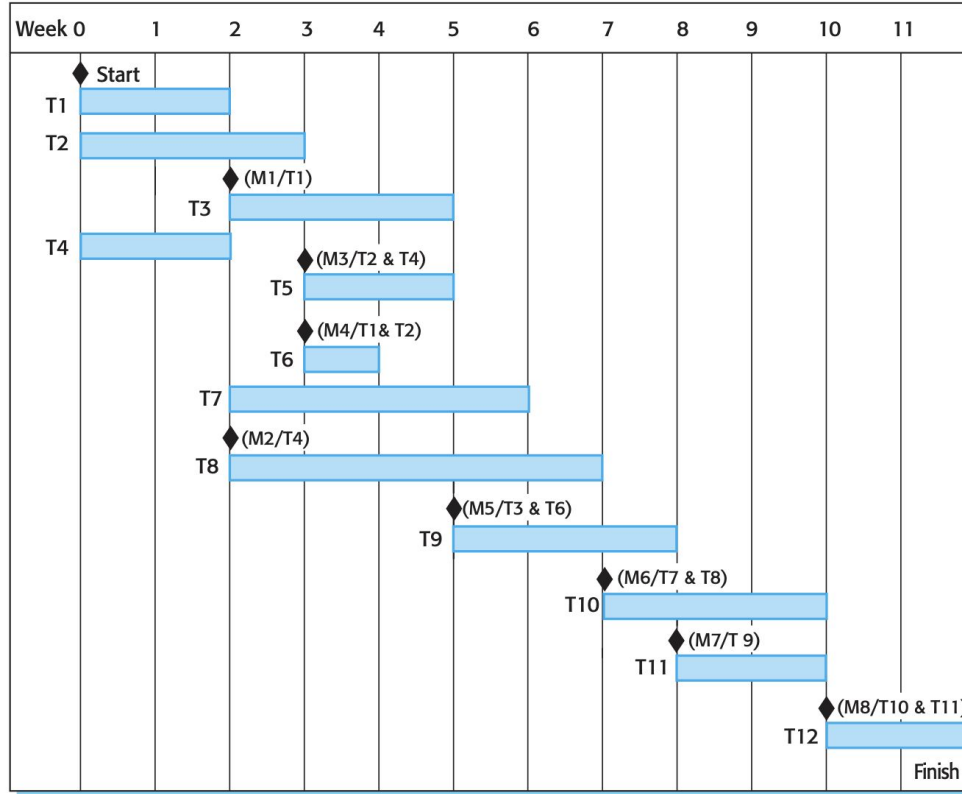


# Tasks, durations, and dependencies

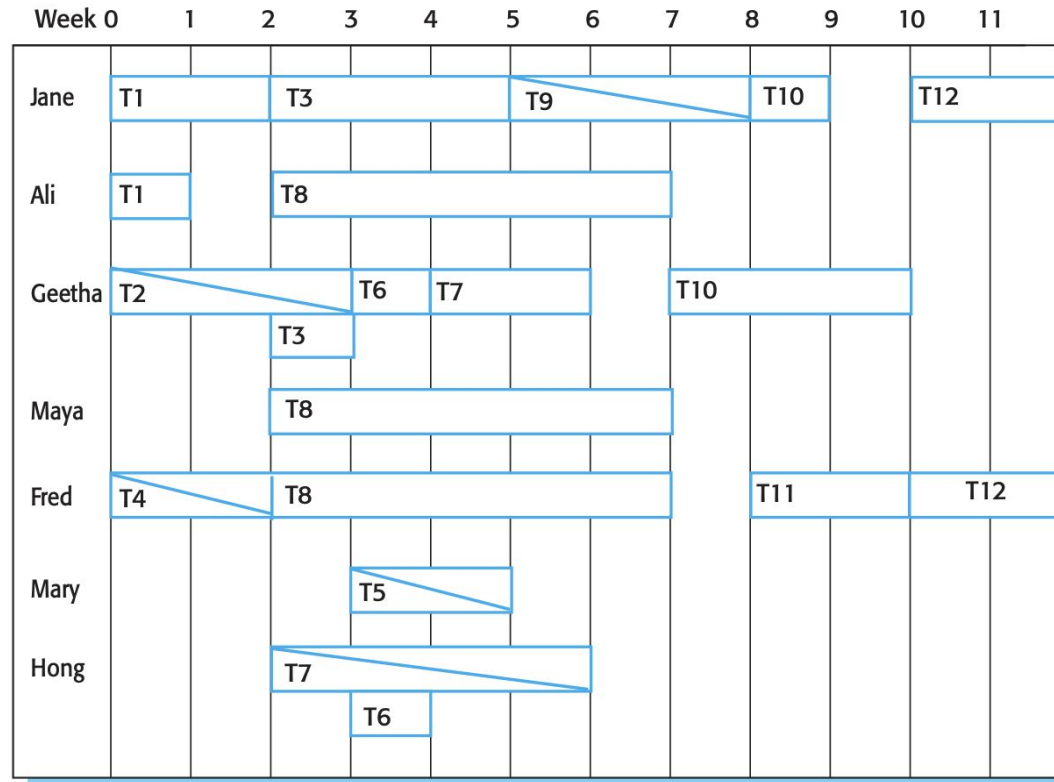
Task	Effort (person-days)	Duration (days)	Dependencies
T1	15	10	
T2	8	15	
T3	20	15	T1 (M1)
T4	5	10	
T5	5	10	T2, T4 (M3)
T6	10	5	T1, T2 (M4)
T7	25	20	T1 (M1)
T8	75	25	T4 (M2)
T9	10	15	T3, T6 (M5)
T10	20	15	T7, T8 (M6)
T11	10	10	T9 (M7)
T12	20	10	T10, T11 (M8)



# Tasks, durations, and dependencies



# Tasks, durations, and dependencies



# Take home study

<https://www.youtube.com/watch?v=sC6UwpVEEE0&t=111s>

Watch this video about one of project management software Trello.

You need to create boards, list, cards for your development project.

Your next task is manage your project in trello



# Q & A



Thank  
You!

