INDIAN INSTITUTE OF INFORMATION TECHNOLOGY

Dharwad
ज्ञानेन विकासः

System Requirements, Processes, Logistics, and Risks

Dr. Animesh Chaturvedi

Assistant Professor: IIIT Dharwad

Young Researcher Alumni: Heidelberg Laureate Forum

Postdoc: King's College London & The Alan Turing Institute

PhD: IIT Indore MTech: IIITDM Jabalpur

HEIDELBERG LAUREATE FORUM

KING'S College LONDON

Indian Institute of Technology Indore
भारतीय प्रौद्योगिकी संस्थान इंदौर

The Alan Turing Institute

PDPM
Indian Institute of Information Technology, Design and Manufacturing, Jabalpur

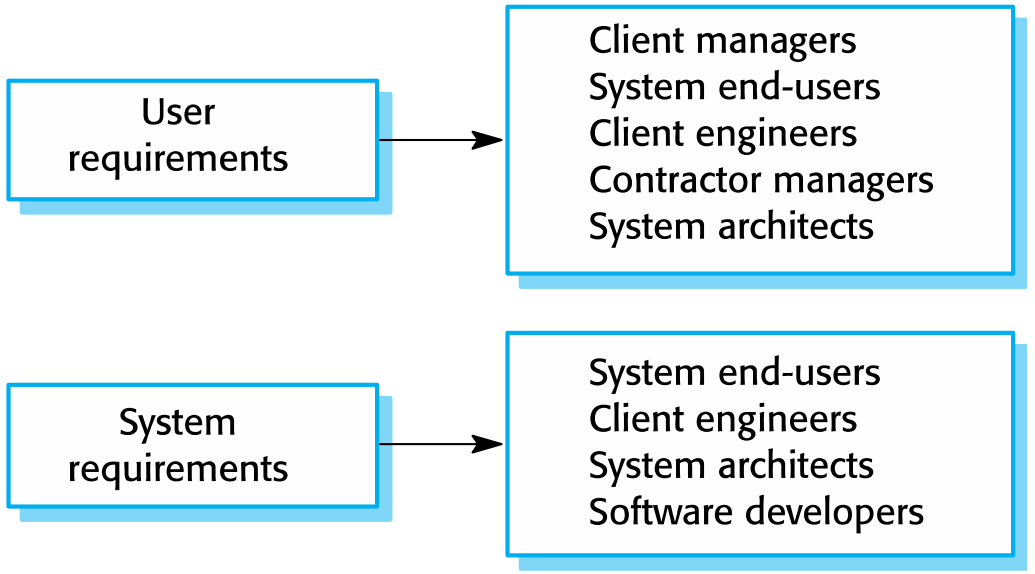
System Requirements Engineering

- Requirements Engineering,
- Requirements specification
- Engineering Design Process and Process Engineering,
- Logistics Management, and
- Risk management

Requirements Engineering

System Requirements

- Requirements describing the functions to satisfy the stakeholder needs and requirements,
- Expressed in textual statements, views, and non-functional requirements; e.g. levels of safety, security, reliability, etc.
- Elicitation (collecting intelligence information) of stakeholder requirements



https://www.sebokwiki.org/wiki/System_Requirements

System Requirements (ISO)

- "A requirement is a statement that identifies a product or processes operational, functional, or design characteristic or constraint, which is unambiguous, testable, or measurable and necessary for product or process acceptability" (ISO 2007)
  - **Process Role or State**:
    - its position in the system block (e.g. translated, derived, satisfied)
    - or its state of agreement (e.g. proposed, approved, cancelled).
  - **Level of Abstraction**: stakeholder requirement, system requirement, system entity requirement
  - **Type of Requirement**: functional, performance, constraint, etc.

https://www.sebokwiki.org/wiki/System_Requirements

System Requirement Types

- **Functional Requirements:** qualitatively the system functions or tasks to be performed in operation
- **Performance Requirements:** quantitatively, or how well and under what conditions a function or task is to be performed (e.g. rates, velocities)
- **Usability Requirements:** Quality of system use (e.g. measurable effectiveness, efficiency, and satisfaction criteria)
- **Interface Requirements:** How the system is required to interact with external systems (external interface), or how system entities interact with each other (internal interface)

https://www.sebokwiki.org/wiki/System_Requirements

System Requirement Types

- **Operational Requirements:** Conditions or properties required for the system to operate
- **Modes or States Requirements:** Events for transitions of modes or states or versions.
- **Adaptability Requirements:** Potential extension, growth, or scalability during the life of the system.
- **Logistical Requirements:** Conditions needed by the continuous utilization. Includes: sustainment (provision of facilities, level support, support personnel, spare parts, training, technical documentation, etc.), packaging, handling, shipping, transportation.

https://www.sebokwiki.org/wiki/System_Requirements

System Requirement Constraints

- **Physical Constraints:** Constraints on weight, volume, and dimension
- **Design Constraints:** Limits on the options that are available to a designer of a solution by imposing immovable boundaries and limits
- **Environmental Constraints:** Environmental conditions to be encountered by the system in its different operational modes.
- **Policies and Regulations Constraints:** Relevant and applicable organizational policies or regulatory requirements that could affect the operation or performance of the system
- Cost and Schedule Constraints

https://www.sebokwiki.org/wiki/System_Requirements

System Requirement Constraints

- the system shall incorporate a legacy or provided system entity,
- certain data shall be maintained in an online repository,
- threats to societal environment (e.g. political, economic, social, etc.),
- natural environment (e.g. wind, rain, temperature, dust, radiation, etc.)
- induced and/or self-induced environmental effects (e.g. motion, shock, noise, electromagnetism, thermal, etc.),
- labor policies, reports to regulatory agency,
- health or safety criteria
- the cost of components of the system, and
- the expected delivery date

https://www.sebokwiki.org/wiki/System_Requirements

System Requirement Artifacts

- System Requirements Document
- System Requirements Justification Document (for traceability purpose)
- System Requirements Database, including traceability, analysis, rationale, decisions, and attributes, where appropriate.
- System External Interface Requirements Document (describes the interfaces of the system with external entities of its context of use)

https://www.sebokwiki.org/wiki/System_Requirements

Stakeholder Requirements

- Stakeholder types
  - End users, System managers, System owners, External stakeholders
- Stakeholder requirements are translated from statements of engineering-oriented language to enable proper architecture definition, design, and verification
- Form the basis of
  - system architecture and design activities
  - system integration and verification activities
  - validation and stakeholder acceptance
  - communication between the various technical staff that interact throughout the project

https://www.sebokwiki.org/wiki/System_Requirements

Requirements specification

Requirements specification

- The process of writing down the user and system requirements in a requirements document.
- User requirements have to be understandable by end-users and customers who do not have a technical background.
- System requirements are more detailed requirements and may include more technical information.
- The requirements may be part of a contract for the system development
  - It is therefore important that these are as complete as possible.

Sommerville, Ian. "Software engineering 10th Edition." (2015).

Natural language specification

- Requirements are written as natural language sentences supplemented by diagrams and tables.
- Used for writing requirements because it is expressive, intuitive and universal. This means that the requirements can be understood by users and customers.
- Problems arise when requirements are not precisely stated.
- Ambiguous requirements may be interpreted in different ways.

Sommerville, Ian. "Software engineering 10th Edition." (2015).

Problems with natural language

- Lack of clarity
  - Precision is difficult without making the document difficult to read.
- Requirements confusion
  - Functional and non-functional requirements tend to be mixed-up.
- Requirements amalgamation
  - Several different requirements may be expressed together.

Sommerville, Ian. "Software engineering 10th Edition." (2015).

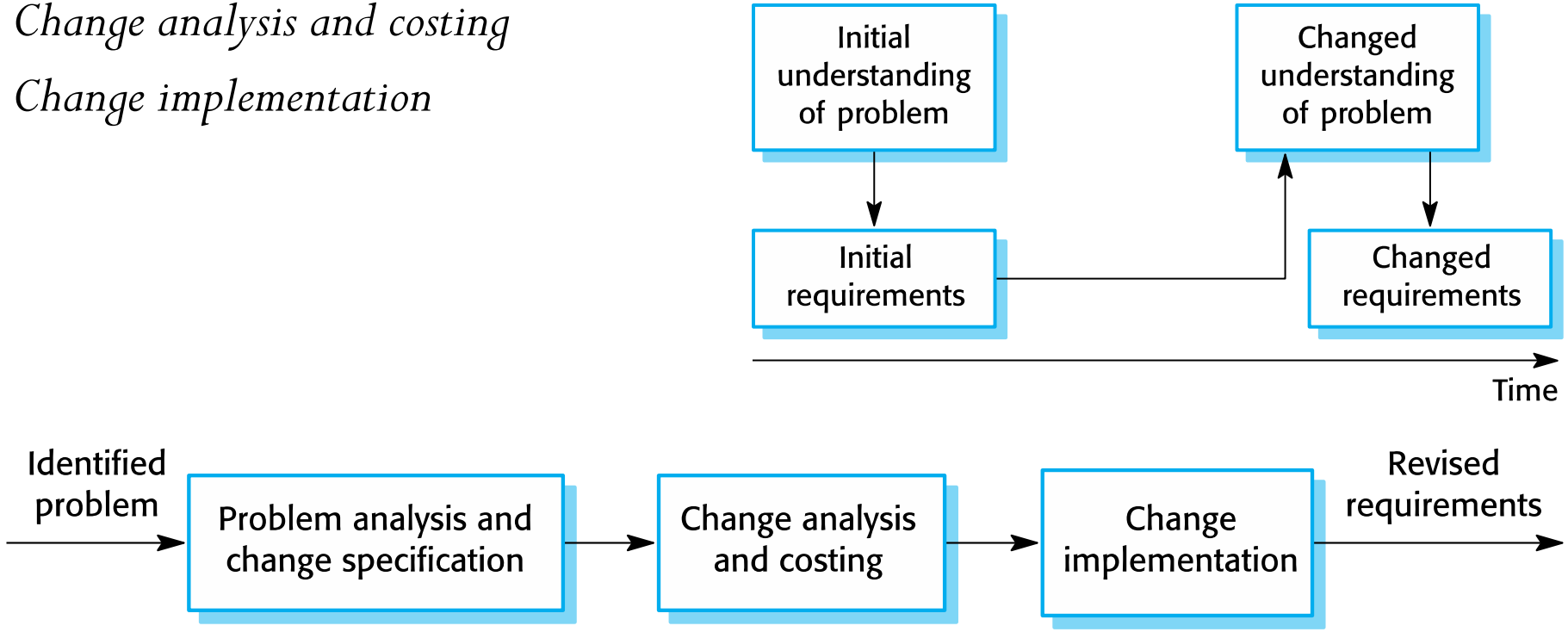
Guidelines for writing requirements

- Invent a standard format and use it for all requirements.
- Use language in a consistent way.
- Use text highlighting to identify key parts of the requirement.
- Avoid the use of systems jargon that require expertize.
- Include an explanation (rationale) of why a requirement is necessary.

Sommerville, Ian. "Software engineering 10th Edition." (2015).

Requirements evolution and change management

- Deciding if a requirements change should be accepted
  - *Problem analysis and change specification*
  - *Change analysis and costing*
  - *Change implementation*



Sommerville, Ian. "Software engineering 10th Edition." (2015).

Summary

- Functional requirements are statements of the services that the system must provide.
- Non-functional requirements often constrain the system. Relate to the emergent properties of the system and therefore apply to the system as a whole.
- Requirements specification is the process of formally documenting the user and system requirements and creating a system requirements document.
- Requirements validation is the process of checking the requirements for validity, consistency, completeness, realism and verifiability. (System testing)
- Business, organizational and technical changes inevitably lead to changes to the requirements for a system.
- Requirements management is the process of managing and controlling these changes. (System evolution and maintenance)

Sommerville, Ian. "Software engineering 10th Edition." (2015).

Engineering Design Process and Process Engineering

Engineering design process

- Series of steps that engineers use in creating functional products and processes.
- The process is highly iterative - parts of the process often need to be repeated many times before another can be entered.
- Decision making process (often iterative) to optimize resources for meeting requirements.
- Feasibility: an evaluation and analysis of the potential of project can proceed into the project design phase

https://en.wikipedia.org/wiki/Engineering_design_process

Requirements engineering processes

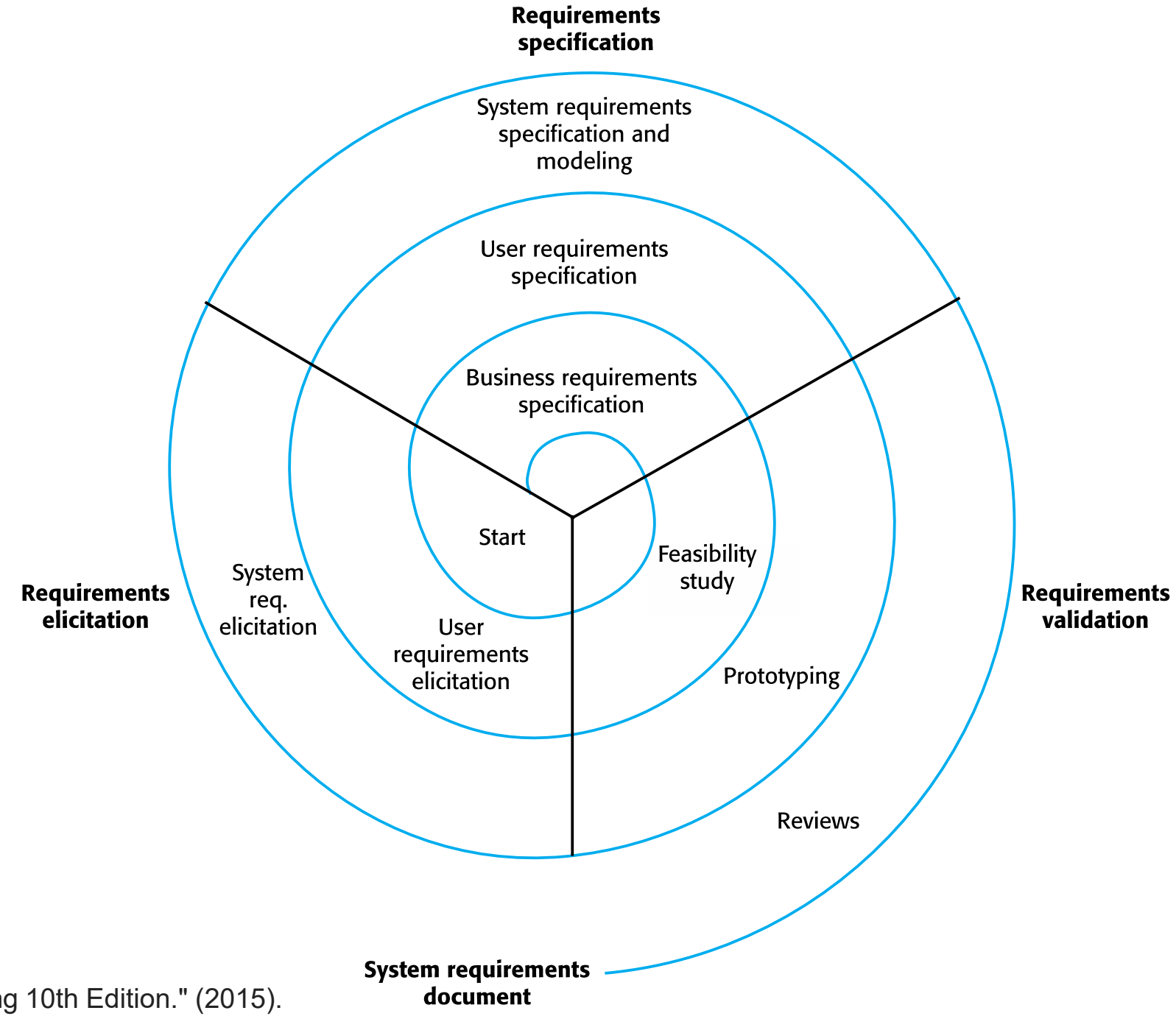
- The processes used for RE vary widely depending on the application domain, the people involved and the organisation developing the requirements.
- However, there are a number of generic activities common to all processes
  - Requirements elicitation;
  - Requirements analysis;
  - Requirements validation;
  - Requirements management.
- RE is an iterative activity in which these processes are interleaved.

Sommerville, Ian. "Software engineering 10th Edition." (2015).

Requirements engineering process

- It is an iterative process that includes requirements elicitation, specification and validation.
- Requirements elicitation is an iterative process that can be represented as a spiral of activities – requirements discovery, requirements classification and organization, requirements negotiation and requirements documentation.
- Requirements elicitation techniques includes interviews and ethnography. User stories and scenarios may be used to facilitate discussions.

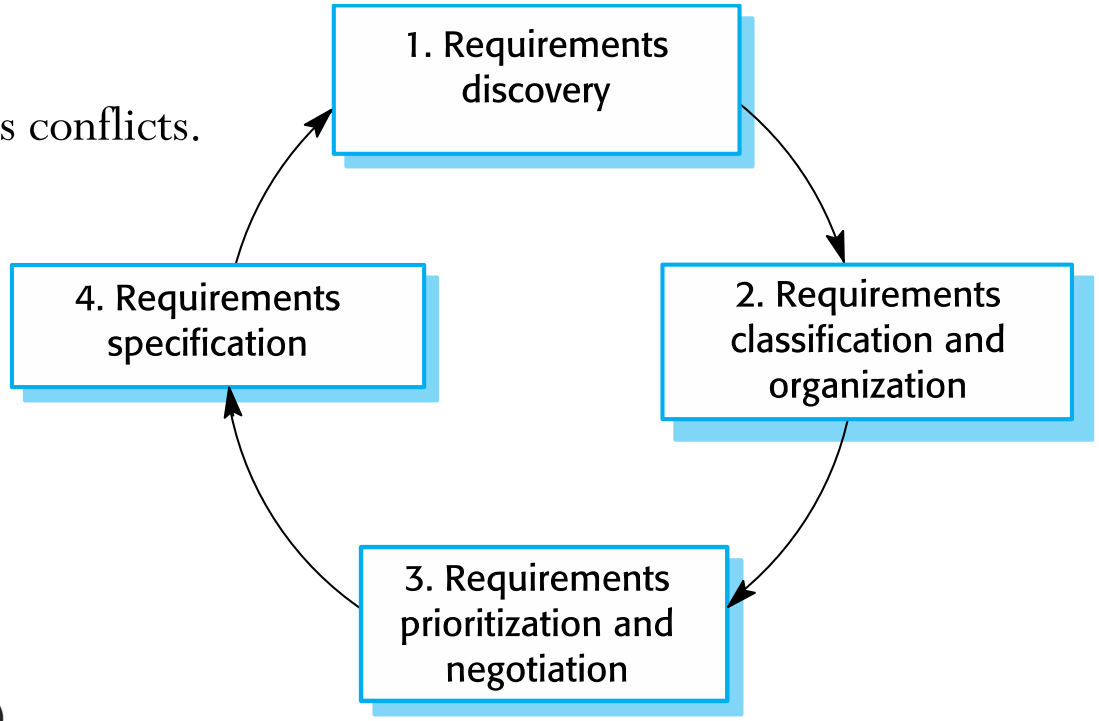
Sommerville, Ian. "Software engineering 10th Edition." (2015).



Sommerville, Ian. "Software engineering 10th Edition." (2015).

Process activities

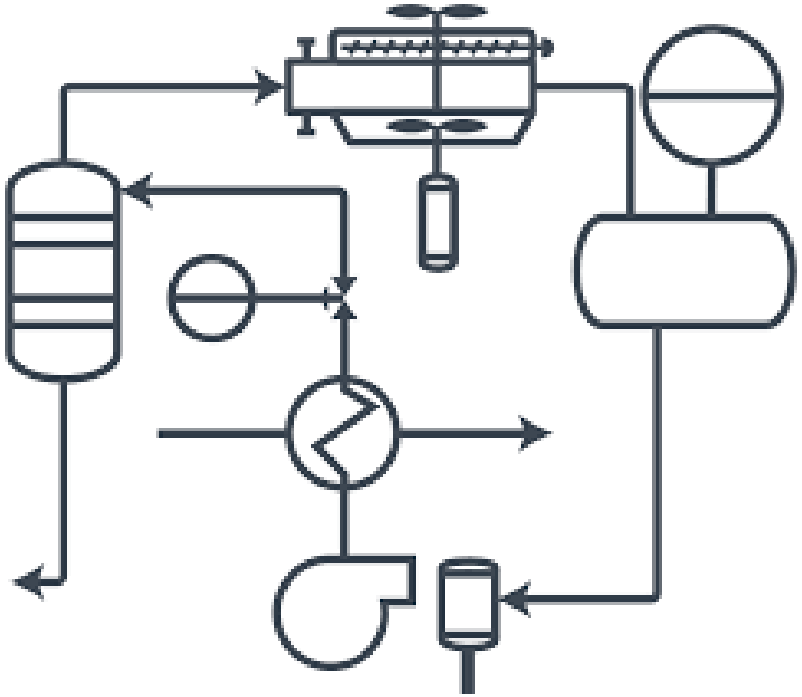
- Requirements discovery
  - Interacting with stakeholders to discover their requirements. Domain requirements are also discovered at this stage.
- Requirements classification and organisation
  - Groups related requirements and organises them into coherent clusters.
- Prioritisation and negotiation
  - Prioritising requirements and resolving requirements conflicts.
- Requirements specification
  - Requirements are documented and input into the next round of the spiral.



Sommerville, Ian. "Software engineering 10th Edition." (2015).

Process Engineering

- Understanding the fundamental principles and laws of engineering processes to transform raw into products useful to society and industry.
- Focuses on the design, control, operation, optimization and intensification of processes.



https://en.wikipedia.org/wiki/Process_engineering

Process Engineering

- Process design: synthesis of graphs or networks, hierarchical decomposition flow sheets, structure optimization, design of the product for the production.
- Process control: model predictive control, controllability measures, robust control, nonlinear control, statistical process control, process monitoring, a collection of measurements, method of taking measurements, and controlling the desired measurement.
- Process operations: scheduling process networks, time-variant planning and optimization, data reconciliation, real-time optimization, flexibility measures, fault diagnosis
- Process Economics: simulation software to find out the break even point, net present value, marginal sales, marginal cost,

https://en.wikipedia.org/wiki/Process_engineering

Process Engineering

- Process Data Analytics: Applying data analytics and machine learning methods for processes of engineering.
- Supporting tools:
  - sequential modular simulation, equation-based process simulation,
  - AI/expert systems, large-scale nonlinear programming,
  - optimization of differential algebraic equations (DAEs), mixed-integer nonlinear programming (MINLP), global optimization, optimization under uncertainty, and
  - quality function deployment (QFD),
  - cost estimation with ASPEN, Super-Pro

https://en.wikipedia.org/wiki/Process_engineering

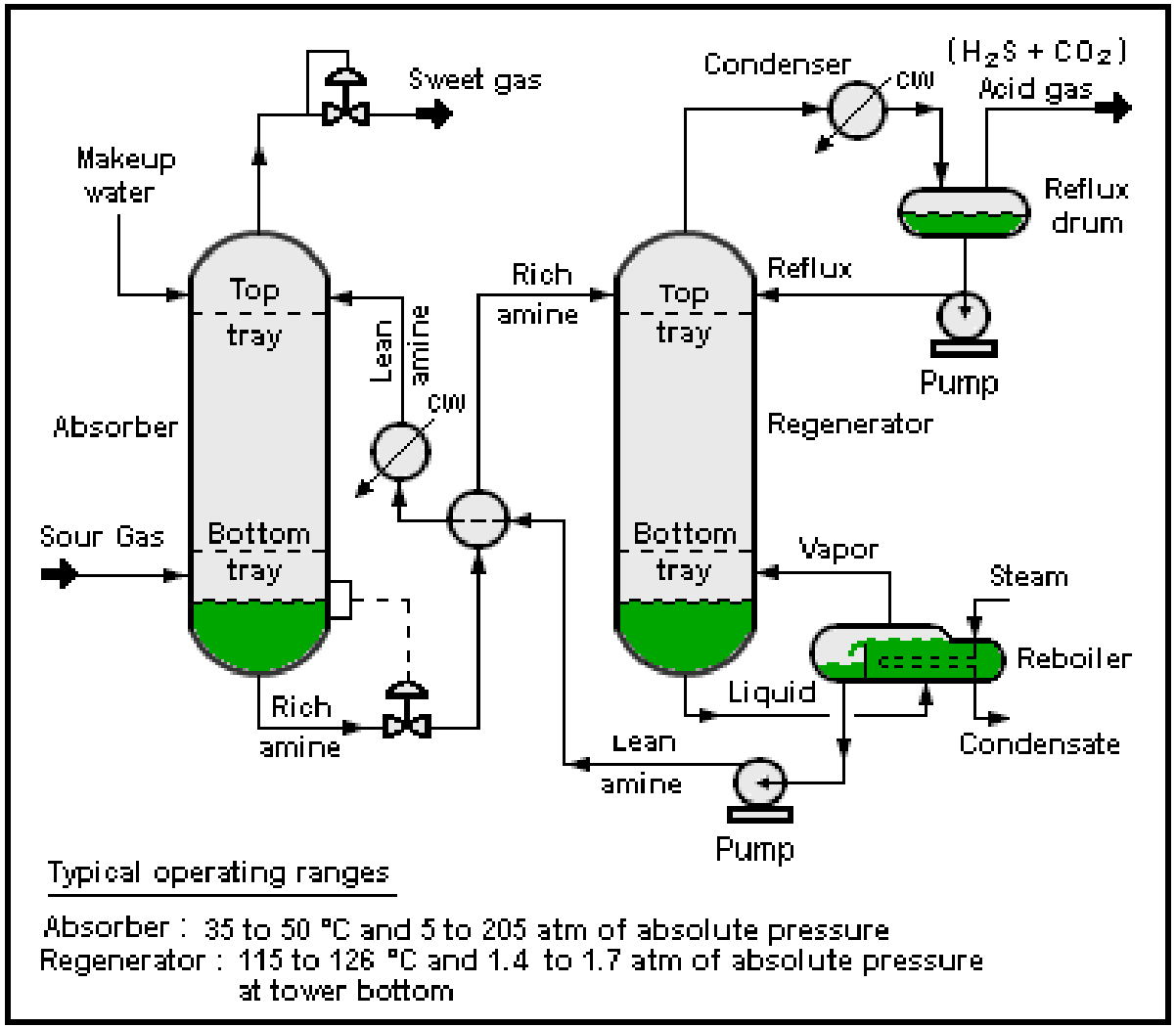
Process Flow Diagram (PFD)

- Process piping
- Major equipment items
- Connections with other systems
- Major bypass and recirculation (recycle) streams
- Operational data (temperature, pressure, mass flow rate, density, etc.), often by stream references to a mass balance.
- Process stream names

https://en.wikipedia.org/wiki/Process_flow_diagram

Process Flow Diagram (PFD)

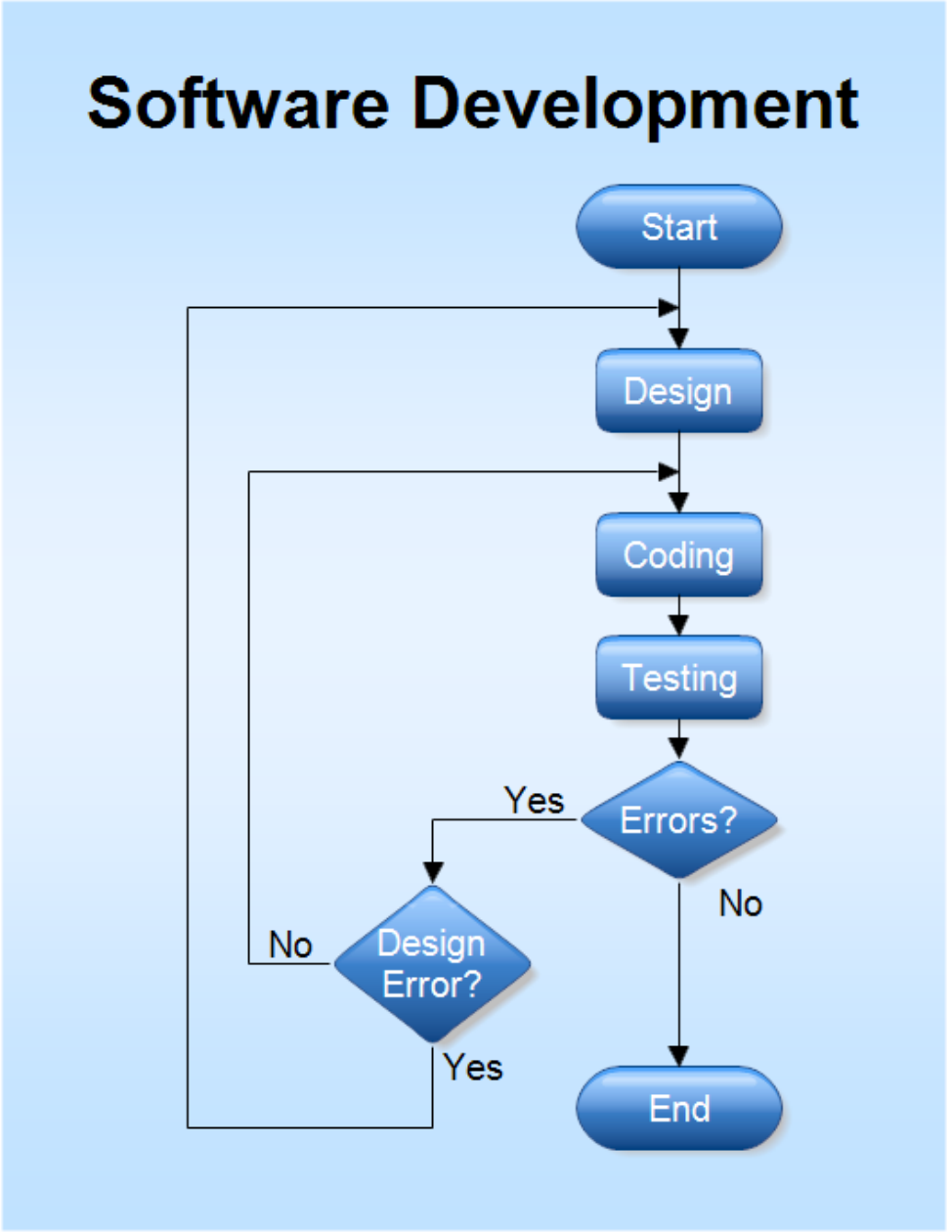
- Flow diagram of a typical amine treating process used in industrial plants



https://en.wikipedia.org/wiki/Process_flow_diagram

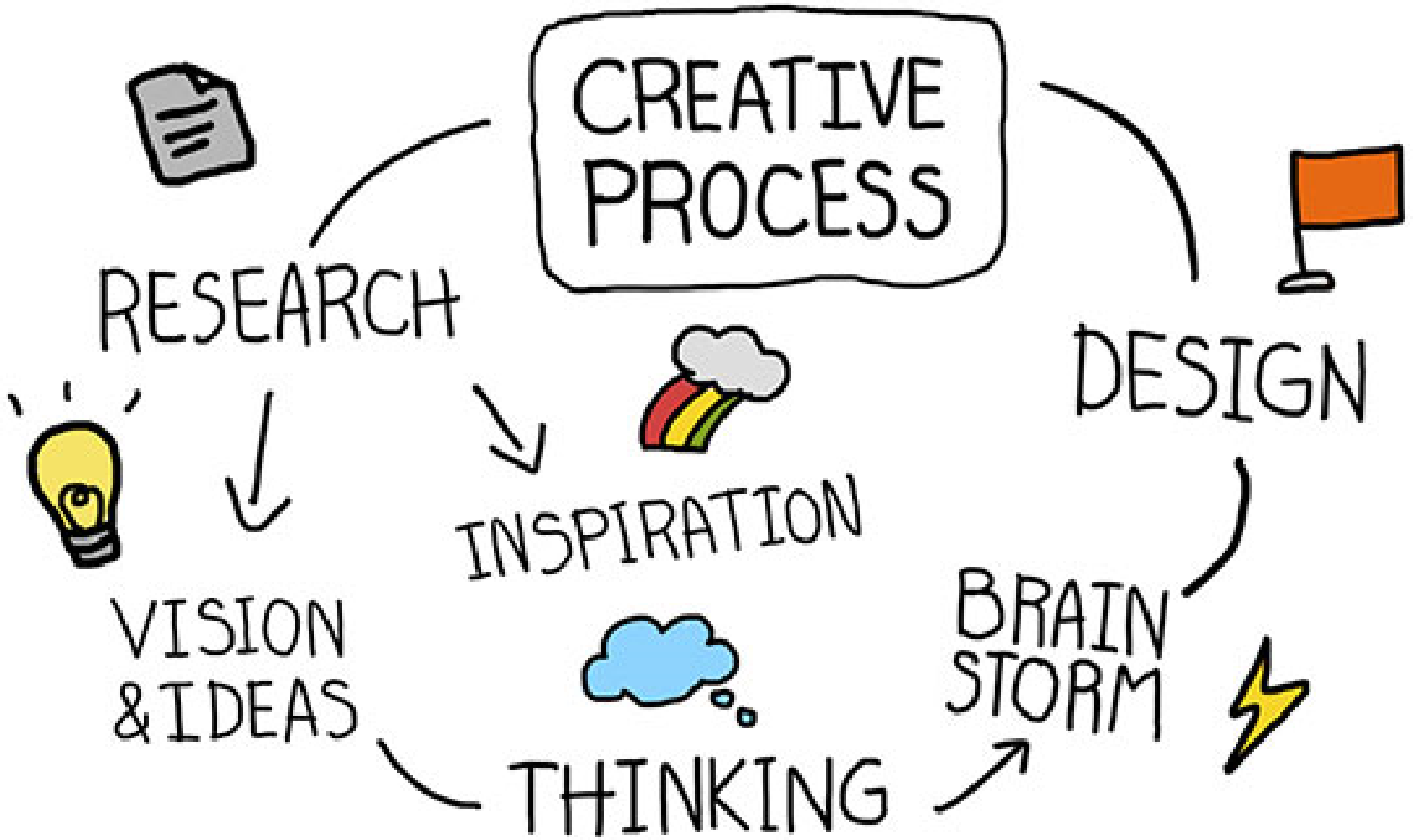
Process Flow Diagram (PFD)

- Flowchart software development cycle.
- Design the system, code it and test it.
- When an error is found,
  - it must be determined whether the error is a design error or not.
    - coding errors can quickly be fixed,
    - but design errors may take longer



https://www.rff.com/software_development.php

Process Flow Diagram (PFD)



http://tonyjuliendesign.com/blog/the-design-process-step-1/

System modeling

- System modeling is the process of developing abstract models of a system, with each model presenting a different view or perspective of that system.
- System modeling has now come to mean representing a system using some kind of graphical notation, which is now almost always based on notations in the Unified Modeling Language (UML).
- System modelling helps the analyst to understand the functionality of the system and models are used to communicate with customers.

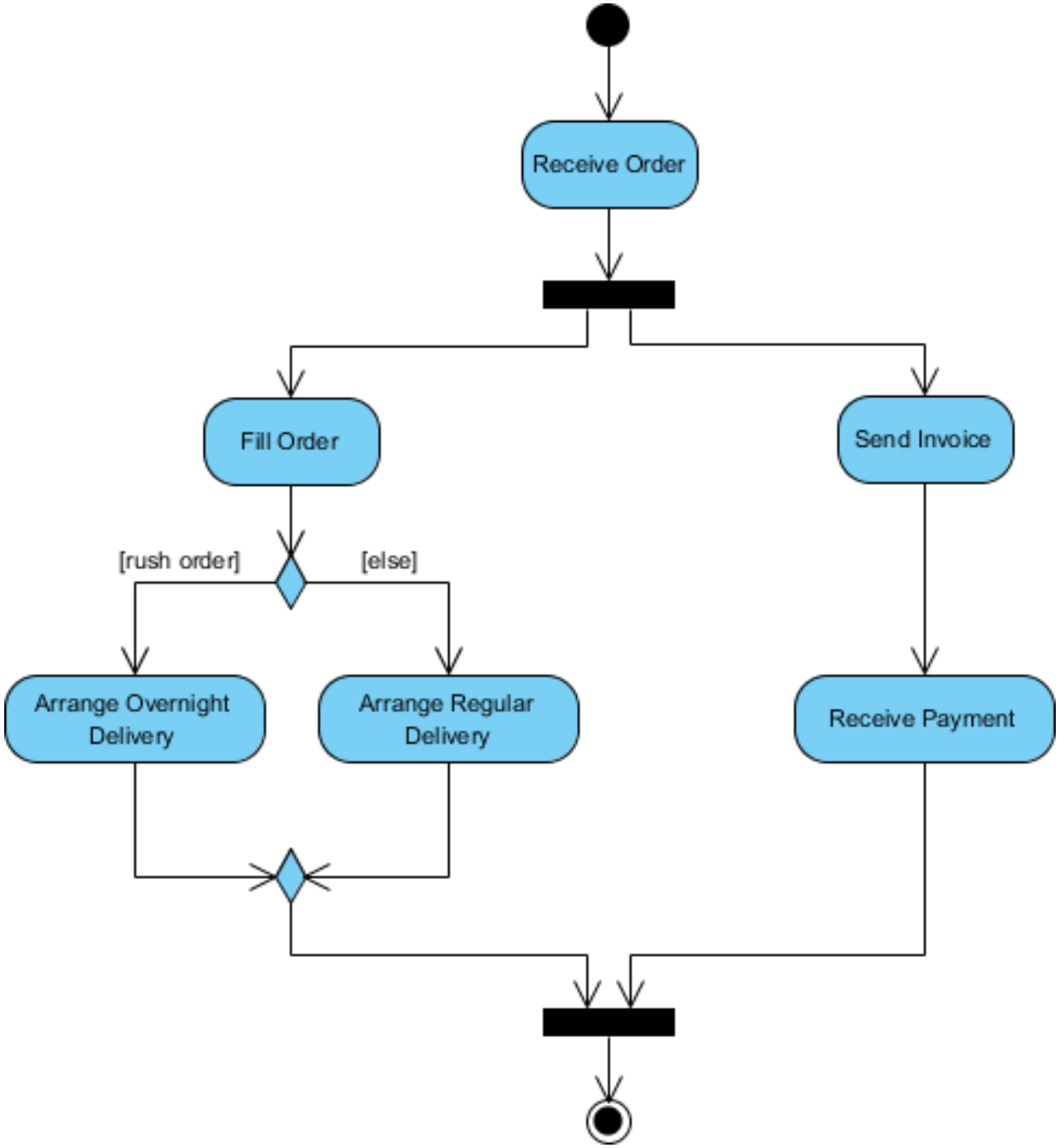
Sommerville, Ian. "Software engineering 10th Edition." (2015).

UML diagram types

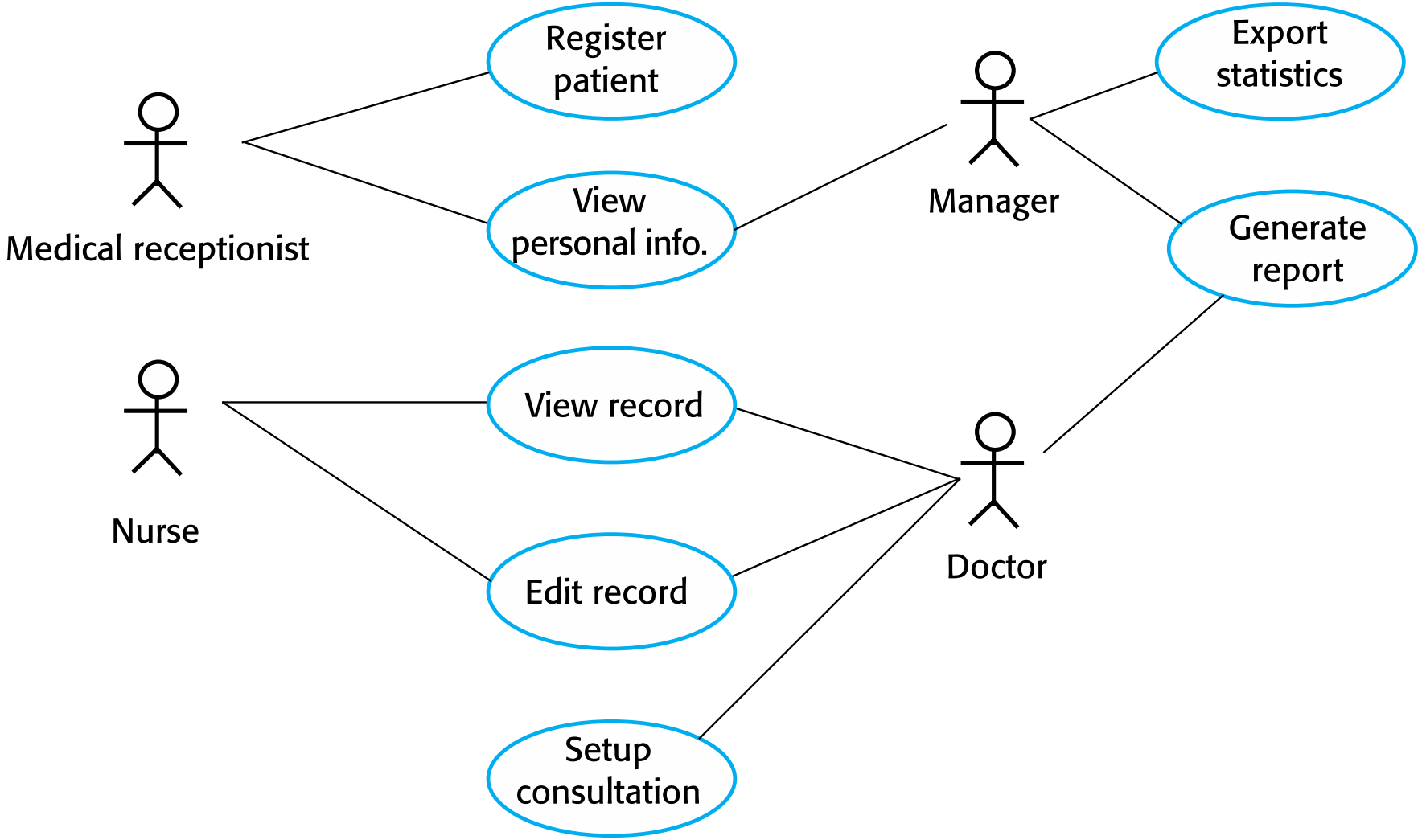
- **Activity diagrams**, which show the activities involved in a process or in data processing .
- **Use case diagrams**, which show the interactions between a system and its environment.
- **Sequence diagrams**, which show interactions between actors and the system and between system components.
- **State diagrams**, which show how the system reacts to internal and external events.

Sommerville, Ian. "Software engineering 10th Edition." (2015).

Activity diagram

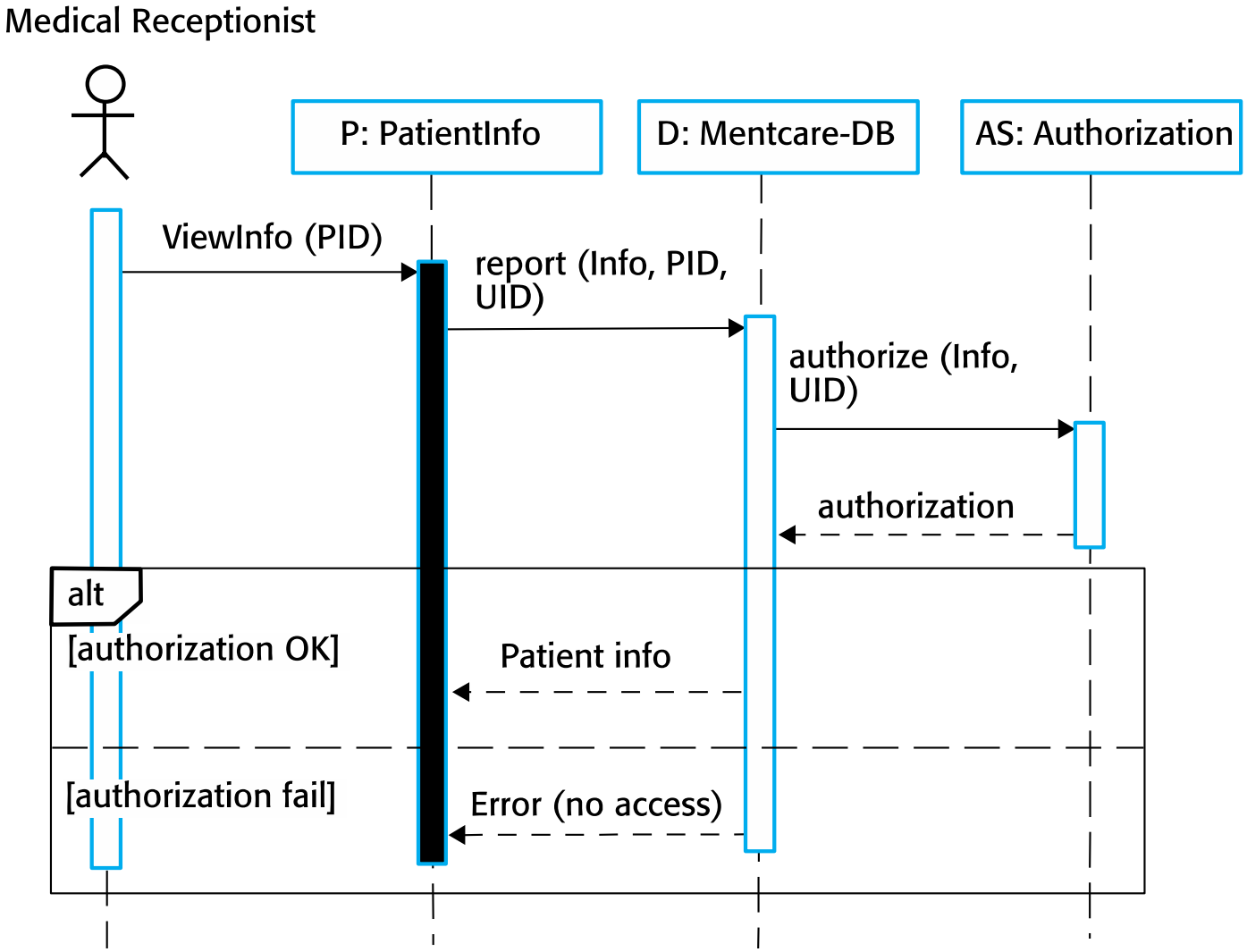


Use cases for the Mentcare system



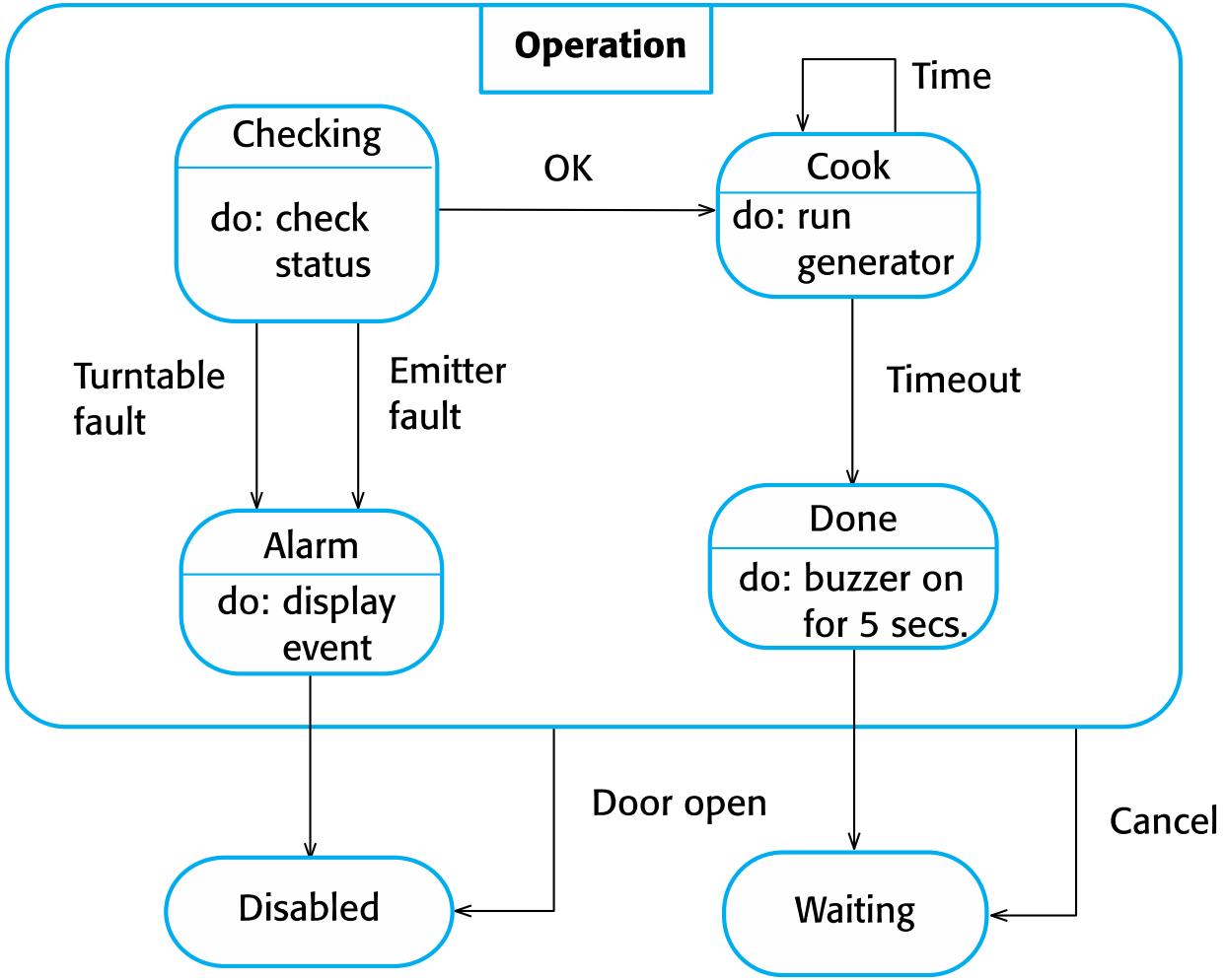
Sommerville, Ian. "Software engineering 10th Edition." (2015).

Sequence diagram for View patient information



Sommerville, Ian. "Software engineering 10th Edition." (2015).

State diagram of a Microwave oven operation



Sommerville, Ian. "Software engineering 10th Edition." (2015).

Logistics Management

Logistics

- It is the science of planning and implementing the acquisition and use of the resources necessary to sustain the operation of a system.
- management of the flow of things between the point of origin and the point of consumption to meet the requirements of customers or corporations
- resources managed includes tangible goods such as
  - Non-perishable goods (like materials, equipment, supplies, etc.)
  - Perishable goods (like food, etc.) and
  - other consumable items
- Example: Military logistics

https://en.wikipedia.org/wiki/Logistics

Logistics

- Ability to *sustain the operation of a system* is determined by the inherent supportability of the system (a function of design) and the processes used to sustain the functions and capabilities of the system in the context of the end user.
- Affects
  - the performance measures: availability, compatibility, interoperability, transportability, reliability, maintainability,
  - the effort and cost: life cycle cost, system operation, maintenance,
  - the environment: manpower, human factors, safety, natural environment effects, habitability.

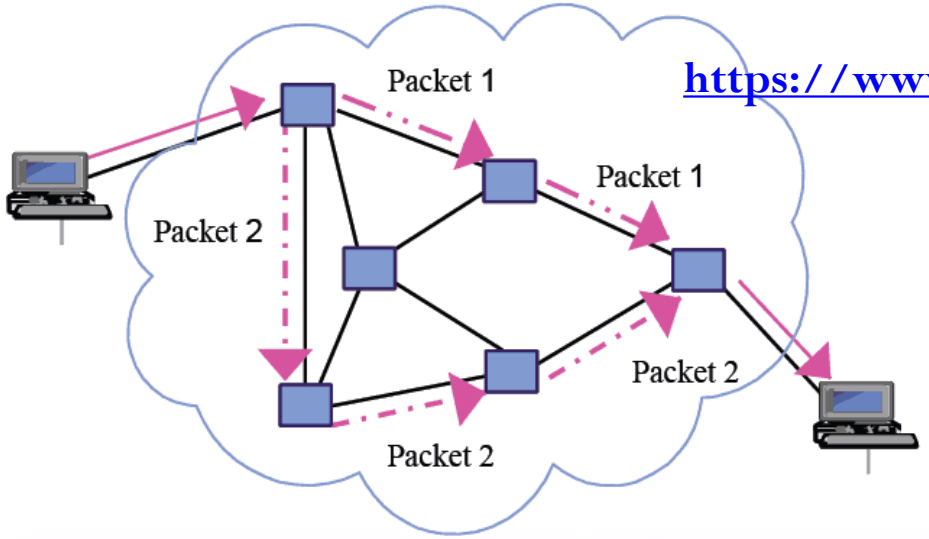
https://www.sebokwiki.org/wiki/Logistics

Logistics management

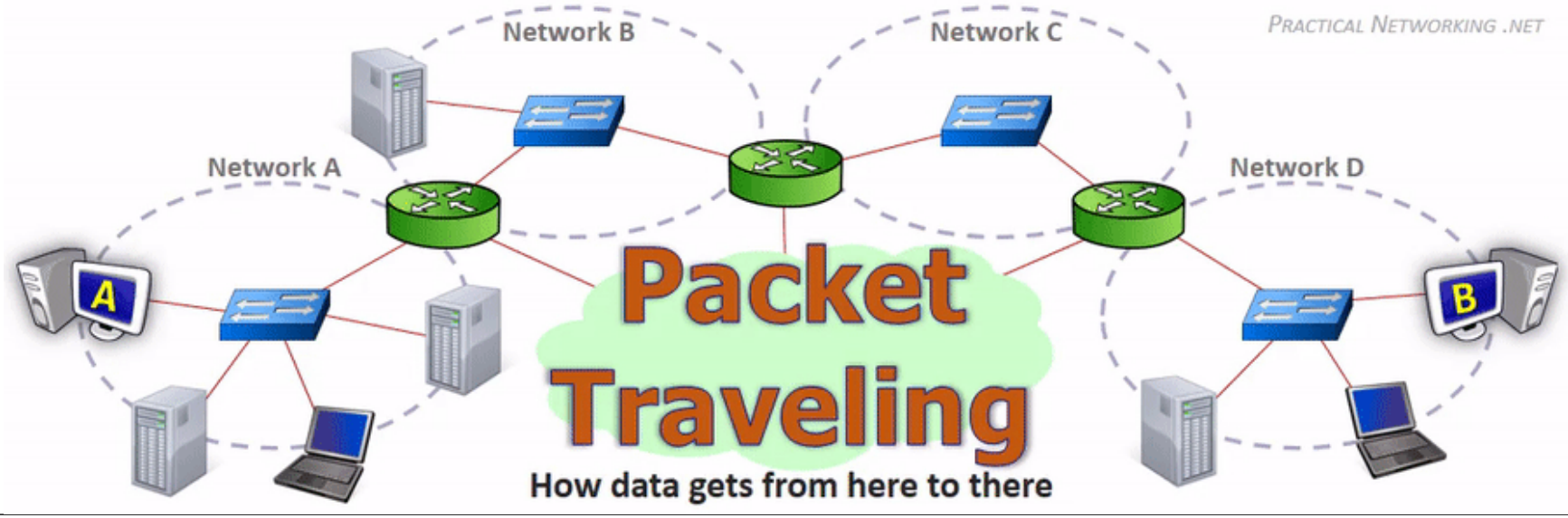
- It plans, implements, and controls the efficient, effective forward, and reverse flow and storage of goods, services, and related information between the point of origin and point of consumption to meet customer's requirements.
- It is the part of
  - supply chain management and
  - supply chain engineering

https://en.wikipedia.org/wiki/Logistics

Computer Network: Packet Transfer



https://www.youtube.com/watch?v=ewrBalT_eBM



Space Transport Logistics

Drone to launch Satellite in space

- **Aevum** believes its **Ravn X drone**, which is said to be the world's biggest **drone**, is now capable of sending low-Earth orbit **satellites** into space
- https://www.youtube.com/watch?v=6YoKuObNPsw

Risk Management

Risk management

- Risk management is concerned with identifying risks and drawing up plans to minimise their effect on a project.
- Project managers assess the risks that may affect a project, monitor these risks and take action when problems arise
  - to anticipate risks,
  - understand the impact of these risks on
    - project,
    - product,
    - business,
  - take steps to avoid these risks

Sommerville, Ian. "Software engineering 10th Edition." (2015).

Risk classification

- There are two dimensions of risk classification
  - The type of risk (technical, organizational, ..)
  - what is affected by the risk:
- *Project risks* affect schedule or resources;
- *Product risks* affect the quality or performance of the system;
- *Business risks* affect the organisation developing or procuring the systems.

Sommerville, Ian. "Software engineering 10th Edition." (2015).

The risk management process

- Risk identification
  - Identify project, product and business risks;
- Risk analysis
  - Assess the likelihood and consequences of these risks;
- Risk planning
  - Draw up plans to avoid or minimise the effects of the risk;
- Risk monitoring
  - Monitor the risks throughout the project;

Sommerville, Ian. "Software engineering 10th Edition." (2015).

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.
  - Organizational risks.
  - People risks.
  - Requirements risks.
  - Estimation risks.

Sommerville, Ian. "Software engineering 10th Edition." (2015).

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.

Sommerville, Ian. "Software engineering 10th Edition." (2015).

Risk planning

- Consider each risk and develop a strategy to manage that risk.
- Avoidance strategies
  - The probability that the risk will arise is reduced;
- Minimization 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;

Sommerville, Ian. "Software engineering 10th Edition." (2015).

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.

Sommerville, Ian. "Software engineering 10th Edition." (2015).

ขอบคุณ
Thai

Grazie
Italian

תודה רבה
Hebrew

धन्यवादः
Sanskrit

ಧನ್ಯವಾದಗಳು
Kannada

Ευχαριστώ
Greek

Thank You
English

Gracias
Spanish

Спасибо
Russian

Obrigado
Portuguese

شكراً
Arabic

https://sites.google.com/site/animeshchaturvedi07

Merci
French

多謝
Traditional
Chinese

धन्यवाद
Hindi

Danke
German

多谢
Simplified
Chinese

நன்றி
Tamil
Tamil

ありがとうございました
Japanese

감사합니다
Korean