

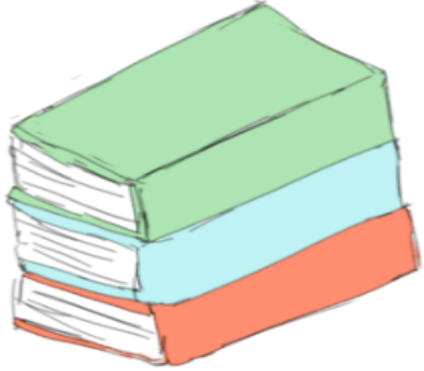
CS 3300
Intro to Software Engineering

SOFTWARE ENGINEERING

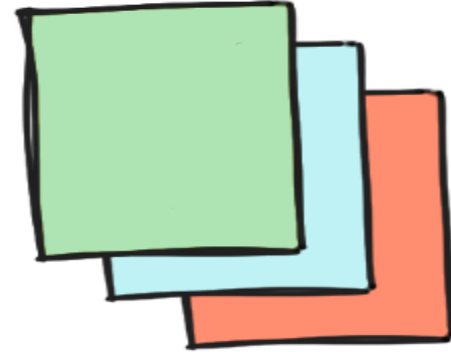
REQUIREMENTS ENGINEERING

Mahdi Roozbahani

LIFECYCLE DOCUMENTS



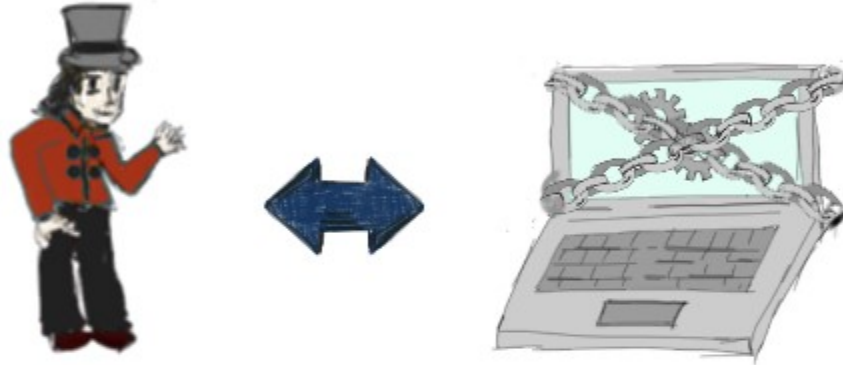
IEEE Documents Definition



Light-weight documents

REQUIREMENTS ENGINEERING (RE)

= =

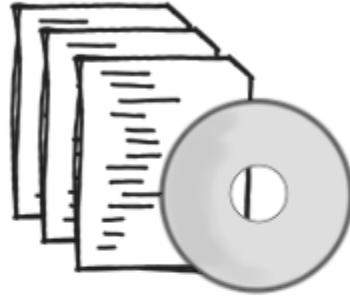


⇒ Software Requirements specification (SRS)



SOFTWARE INTENSIVE SYSTEMS

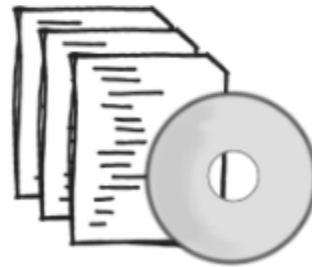
software



software intensive system = software + hardware + Context



=



+



+



SOFTWARE QUALITY



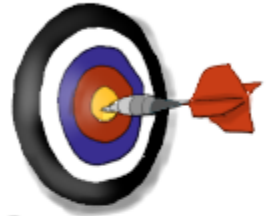
Software

runs on some



hardware

and is developed for a



Purpose

that is related to human activities

$$\text{Quality} = f(\text{Software}, \text{Purpose})$$



Requirements engineering is mostly about identifying the purpose

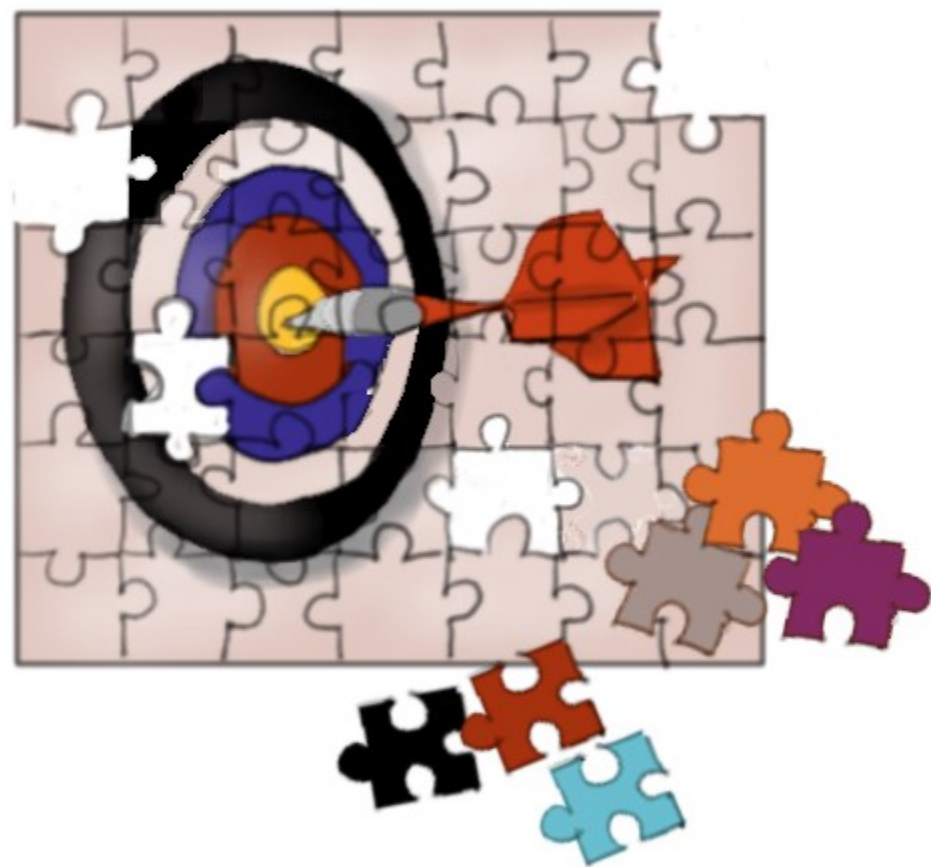
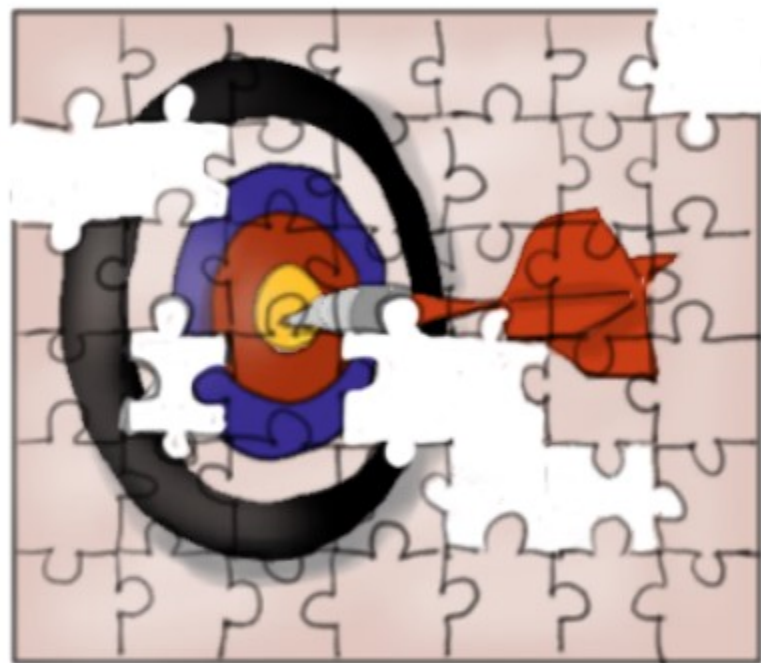
IDENTIFYING PURPOSE = DEFINING REQUIREMENTS



Extremely hard task!

- Sheer complexity of the purpose/requirements
- Often, people don't know what they want until you show it to them
- Changing requirements
- Multiple stakeholders with conflicting requirements

COMPLETENESS AND PERTINENCE





Consider an information system for a gym. In the list below, mark all the requirements that you believe are pertinent

- Members of the gym shall be able to access their training programs
- The system shall be able to read member cards
- The system shall be able to store members' commute time
- Personal trainers shall be able to add clients
- The list of members shall be store as a linked list

Is the above list complete?

- Yes
- No



Why can irrelevant requirements be harmful?

- They may lead to missing functionality in the final product
- They can introduce inconsistency
- They can waste project resources
- They may introduce bugs in the software system

BEST PRACTICE ?



DEFINITION OF REQUIREMENTS ENGINEERING

Not a phase or a stage

Communication is as important as analysis

Quality means fitness-for-purpose. Cannot say anything about quality unless you understand the purpose

Needed to identify all the stakeholders - not just the customer and the user

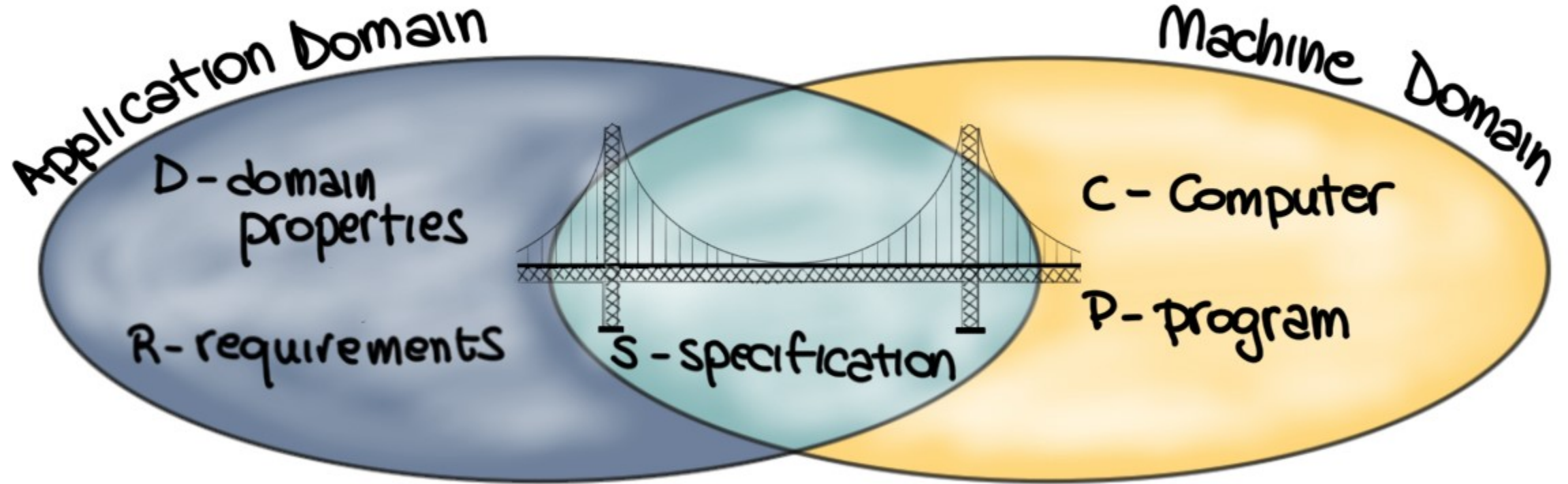
Requirements engineering (RE) is a set of activities concerned with identifying and communicating the purpose of a software-intensive system, and the context in which it will be used. Hence, RE acts as the bridge between the real-world needs of users, customers, and other constituencies affected by a software system, and the capabilities and opportunities afforded by software-intensive technologies

Designers need to know how and where the system will be used

Requirements are partly about what is needed ...

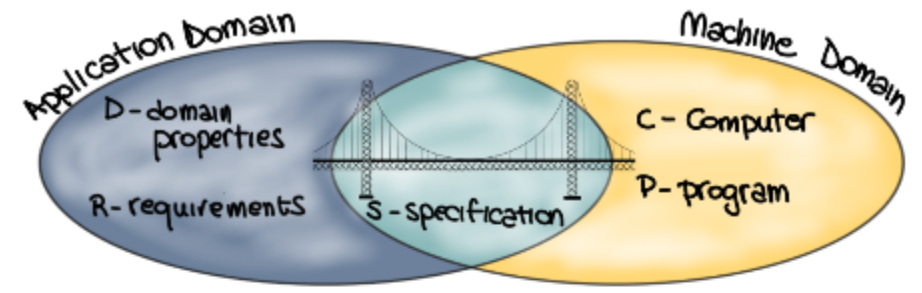
...and partly about what is possible

WHAT ARE REQUIREMENTS?





Referring to the figure that we just discussed, indicate, for each of the following items, whether they belong to the machine domain (1), application domain (2), or their intersection (3) (Enter the corresponding number - 1, 2, or 3 - in the entry next to the item)



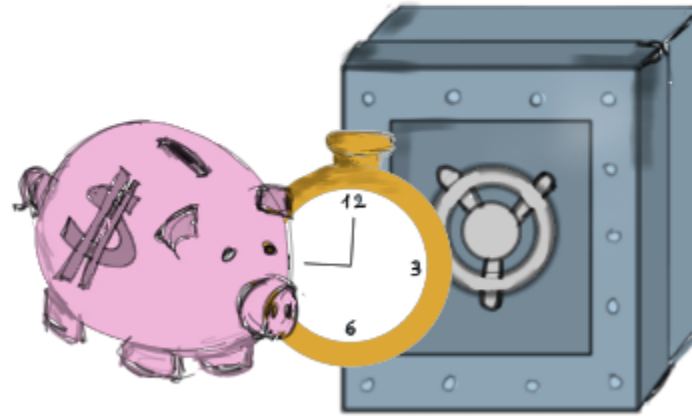
- An algorithm sorts a list of books in alphabetical order by the first author's name
- A notification of the arrival of a message appears on a smart watch
- An employee wants to organize a meeting with a set of colleagues
- A user clicks a link on a web page

FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS

Functional



Non-functional



USER AND SYSTEM REQUIREMENTS



User Requirements

- Written for customers
- often in natural language, no technical details

System requirements

- written for developers
- detailed functional and non-functional requirements
- clearly and more rigorously specified





UserRequirementsDefinition

The software must provide a means of representing and accessing external files created by other tools.



SystemRequirementsDefinition

1.1 The user should be provided with facilities to define the type of external files

1.2 Each external file type may have an associated tool which may be applied to the file.

1.3 Each external file type may be represented as a specification from the user's display.

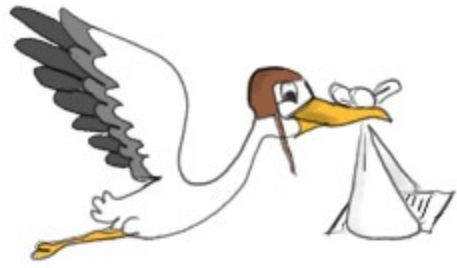
1.4 Facilities should be provided for the icon representing an external file type to be defined by the user.

1.5 When the user selects an icon representing an external file, the effect of that selection is to apply the tool associated with the type of the external file to the file represented by the selected icon.



Which of the following requirements are non-functional requirements?

- The BowlingAlley program keeps track of the score during a game
- The WordCount program should be able to process large files
- The Login program for a website should be secure
- The VendingMachine program should take coins as an input from the user.



WHERE DO REQUIREMENTS COME FROM?

Stakeholders



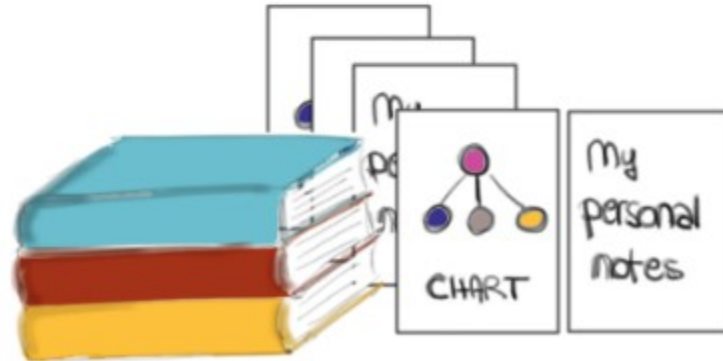
...

Application domain



...

Documentation

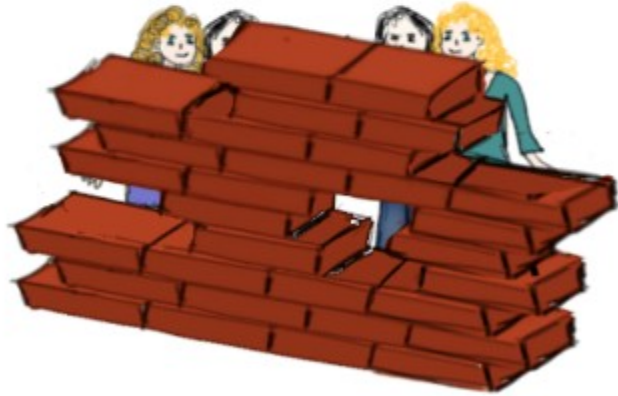


...

ELICITATION PROBLEMS



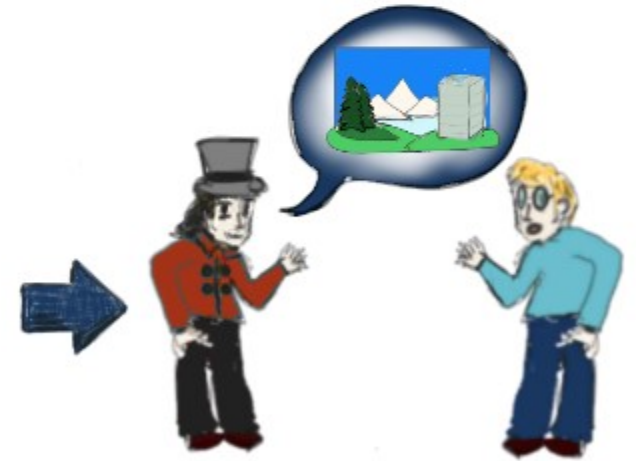
thin spread of domain knowledge



Limited observability



knowledge is tacit



Bias

TRADITIONAL TECHNIQUES



Background reading



Hard data and samples



Interviews



surveys



Meetings

MODELING REQUIREMENTS



Modeling enterprises

- goals and objectives
- organizational structure
- task and dependencies
- agents, roles, intentionality



Organization modeling
i*, soft system modeling ...
Goal modeling
KAOS, CREWS ...

Modeling information and behaviors

- information structure
- behavioral view
 - scenarios and use cases
 - state machine models
 - sequence diagrams
 - information flow
- time / sequencing requirements



Information modeling
E-R, class diagram ...
Structure analysis
Struc. Analysis and design tech. (SADT)
Jackson software development ...
Object oriented analysis
UML
Formal methods
Alloy, Petri Net, Z ...

Modeling system qualities (NFRs)

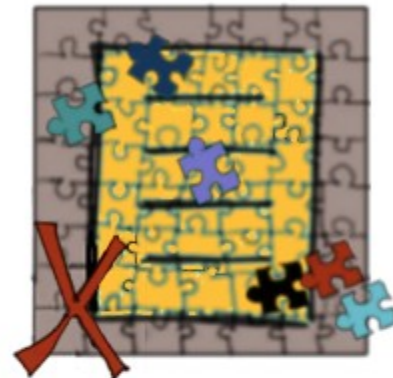
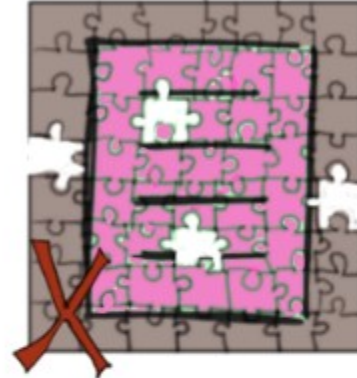
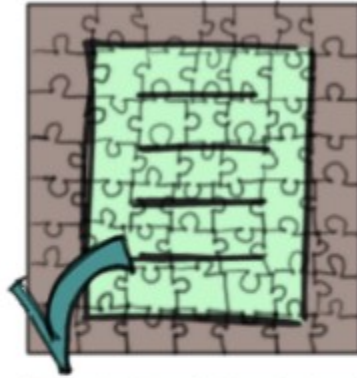


Quality tradeoffs
Win-win, NFR, Analytic Hierarchy
process (AHP) ...
Specific NFRs
Timed Petri net (performance)
Task models (usability) ...

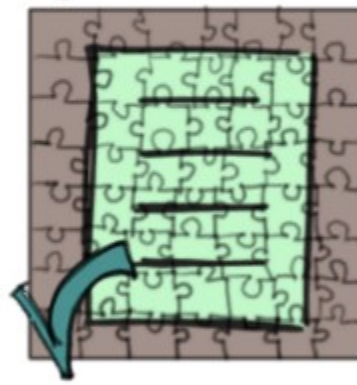
ANALYZING REQUIREMENTS



Verification



Validation

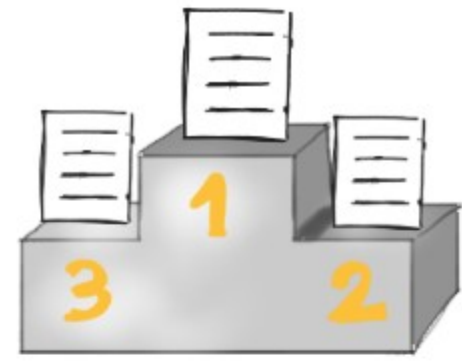


Risk analysis

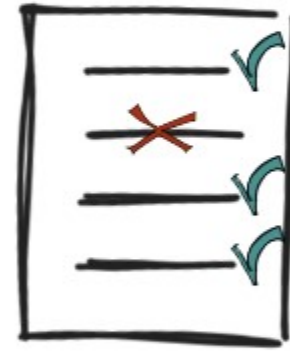


REQUIREMENTS PRIORITIZATION

Limited resources



⇒ inability to satisfy all the requirements



⇒ need to prioritize them

mandatory
nice to have
superfluous





Imagine that you have collected the following set of five requirements for an ATM system but only have resources to satisfy two, possibly three of those.

Suitably prioritize the requirements by marking them as mandatory (M), nice to have (N), or superfluous (S).

- [] The system shall check the PIN of the ATM card before allowing the customer to perform an operation
- [] The system shall perform an additional biometric verification of the customer's identity before it allows the customer to perform an operation
- [] The system shall allow customers to withdraw cash using an ATM card
- [] The system shall allow customers to deposit money using an ATM card
- [] The system shall allow customers to change the PIN of an ATM card

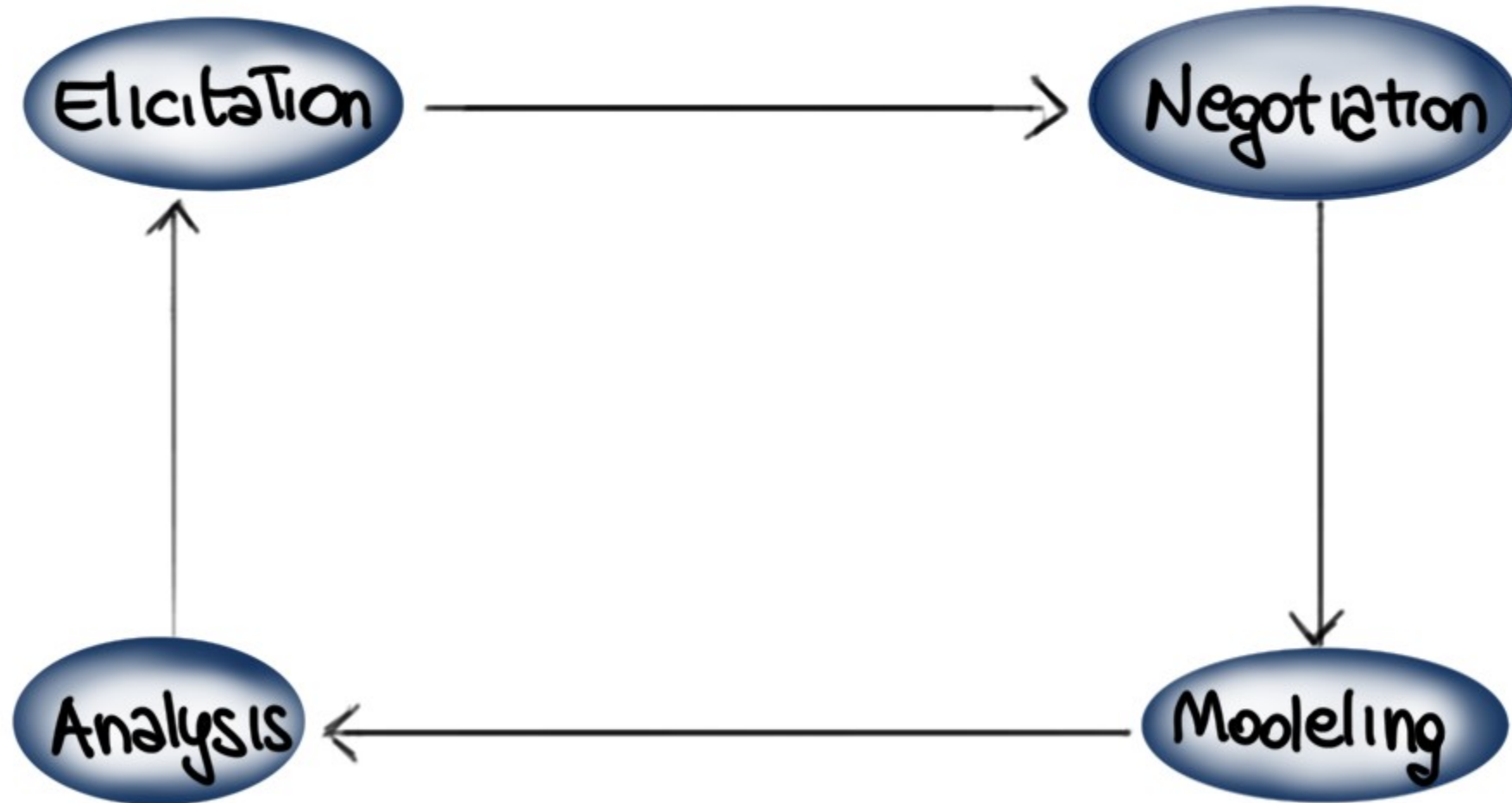
REQUIREMENTS ENGINEERING PROCESS

Elicitation

Analysis

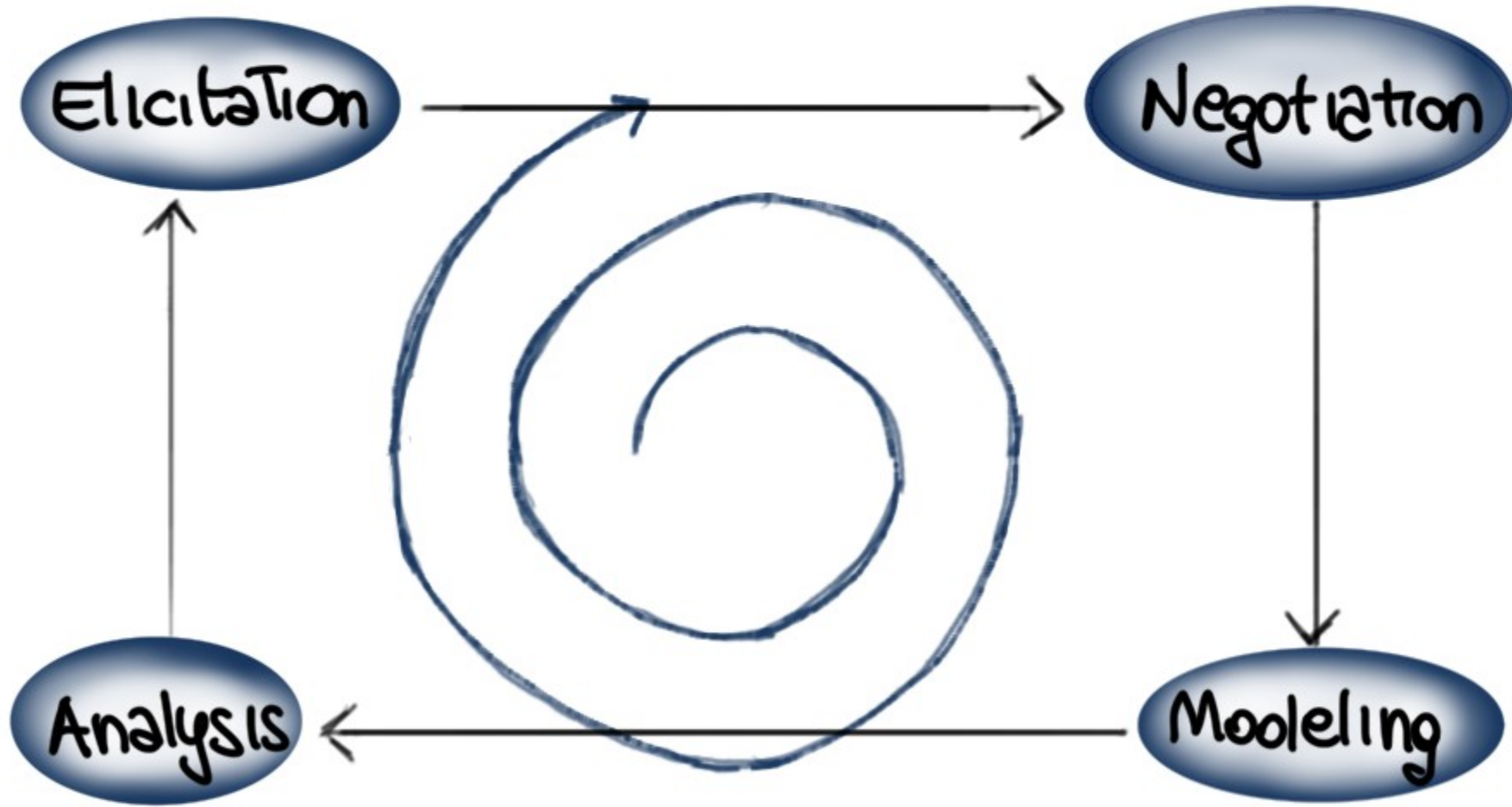
Modeling

REQUIREMENTS ENGINEERING PROCESS



REQUIREMENTS ENGINEERING PROCESS

ITERATIVE



SOFTWARE REQUIREMENTS SPECIFICATION (SRS)



Way to communicate requirements to others



Different projects require different SRSs



Introduction

User requirements

System requirements
(functional and non-functional)