



Software Requirement

An Introduction

Dr. Zeinab Teimoori
SENG 3130
Thompson Rivers University

Objectives

- Why do we need to discover the requirements?
- What are the software requirements?
- Levels and types of requirements
- Product vs Project requirements
- Requirements engineering

Objectives

- Why do we need to discover the requirements?
- What are the software requirements?
- Levels and types of requirements
- Product vs Project requirements
- Requirements engineering

Let's listen to this conversation



Ref: https://www.deepijatel.com/blog/uploads/images/How_to_set_up_a_call_center_in_India-01.png

Maria: “Hello, Phil? This is Maria in Human Resources. We’re having a problem with the personnel system you programmed for us. An employee just changed her name to Sparkle Starlight, and we can’t get the system to accept the name change. Can you help?”

Phil: “She married some guy named Starlight?”

Maria: “No, she didn’t get married, just changed her name. That’s the problem. It looks like we can change a name only if someone’s marital status changes.”

Phil: “Well, yeah, I never thought someone might just change her name. I don’t remember you telling me about this possibility when we talked about the system.”

Maria: “I assumed you knew that people could legally change their name anytime they like. We have to straighten this out by Friday or Sparkle won’t be able to cash her paycheck. Can you fix the bug by then?”

Phil: “It’s not a bug! I never knew you needed this capability. I’m busy on the new performance evaluation system. I can probably fix it by the end of the month, but not by Friday. Sorry about that. Next time, tell me these things earlier and please write them down.”

Maria: “What am I supposed to tell Sparkle? She’ll be upset if she can’t cash her check.”

Phil: “Hey, Maria, it’s not my fault. If you’d told me in the first place that you had to be able to change someone’s name at any time, this wouldn’t have happened. You can’t blame me for not reading your mind.”

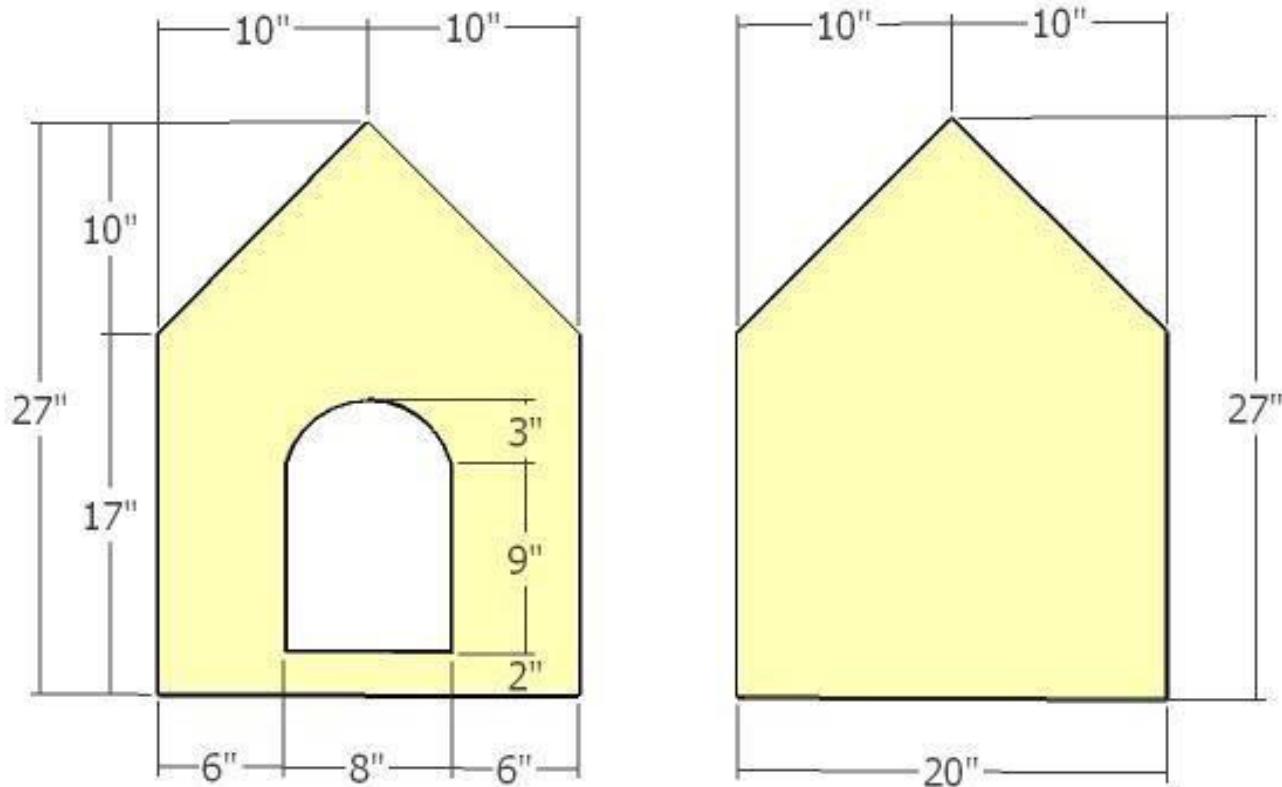
Maria: “Yeah, well, this is the kind of thing
that makes me hate computers. Call me as
soon as you get it fixed, will you?”

It is so easy to build a software system! Just write the code



MyOutdoorPlans.com

Just write the code



MyOutdoorPlans.com

Just write the code



[1] McConnell, Steve. "Code complete", 2nd ed. 2004

It is so easy to build a software system! Just write the code

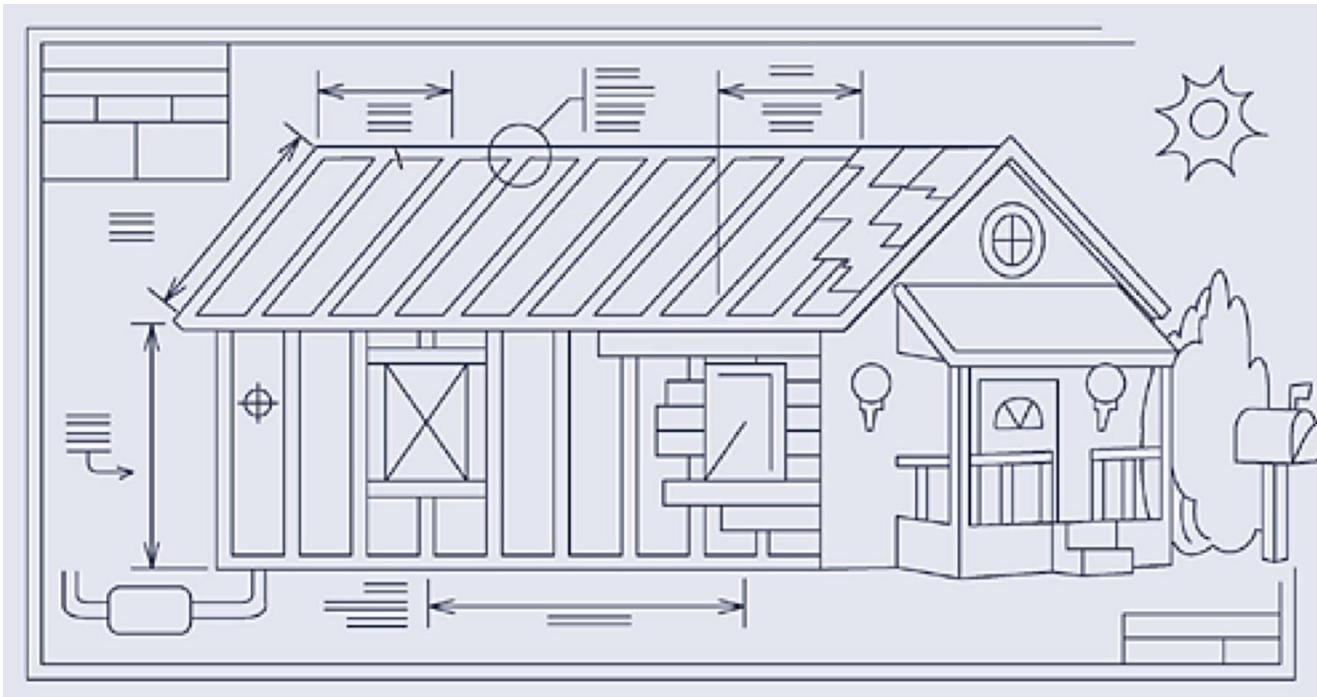


[1]

*"The penalty for a mistake on a **simple structure** is only a little time and maybe some embarrassment"* [1]

[1] McConnell, Steve. "Code complete", 2nd ed. 2004

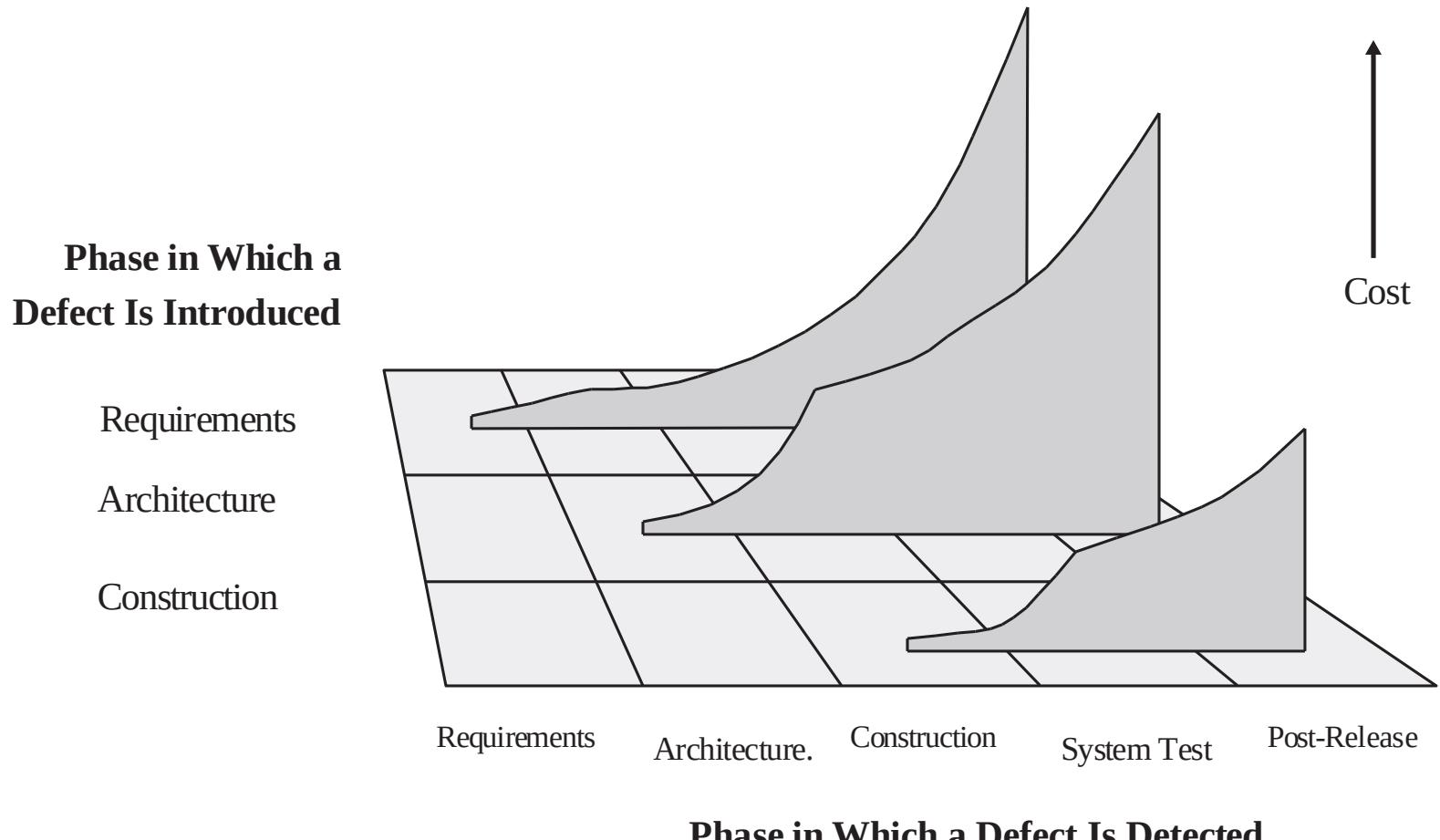
Actual software systems are very complex



[1] In large-scale software systems, **implementing a missing requirement** takes considerable effort.

[1] McConnell, Steve. "Code complete", 2nd ed. 2004

Fixing requirements errors are very costly



[1] McConnell, Steve. "Code complete", 2nd ed. 2004

Single requirement error can lead to a mission failure



- For example, the loss of the “Mars Climate Orbiter”
- According to NASA reports “*The root cause of the loss of the spacecraft was the failed translation of English units into metric units in a segment of ground-based, navigation-related mission software*”

[1]

[1] <https://mars.nasa.gov/msp98/news/.html>

Single requirement error can lead to a mission failure (cont.)



- The process to **verify** and **validate** certain **engineering requirements** and technical interfaces between some project groups, and between the project and its prime mission contractor, was **inadequate**.

[1] <https://mars.nasa.gov/msp98/news/mco991.110.html>

Defects in the software products can destroy your reputation



“Errors introduced during requirements activities account for 40 to 50 percent of all defects found in a software product” [1]

Summary of the importance of software requirements



Determining the success of your product



Economic impact



Impacts your brand reputation

Objectives

- Why do we need to discover the requirements?
- **What are the software requirements?**
- Levels and types of requirements
- Product vs Project requirements
- Requirements engineering

What makes a successful project?

- Think about the best software that you like and tell us why you liked it.

What are the software requirements?

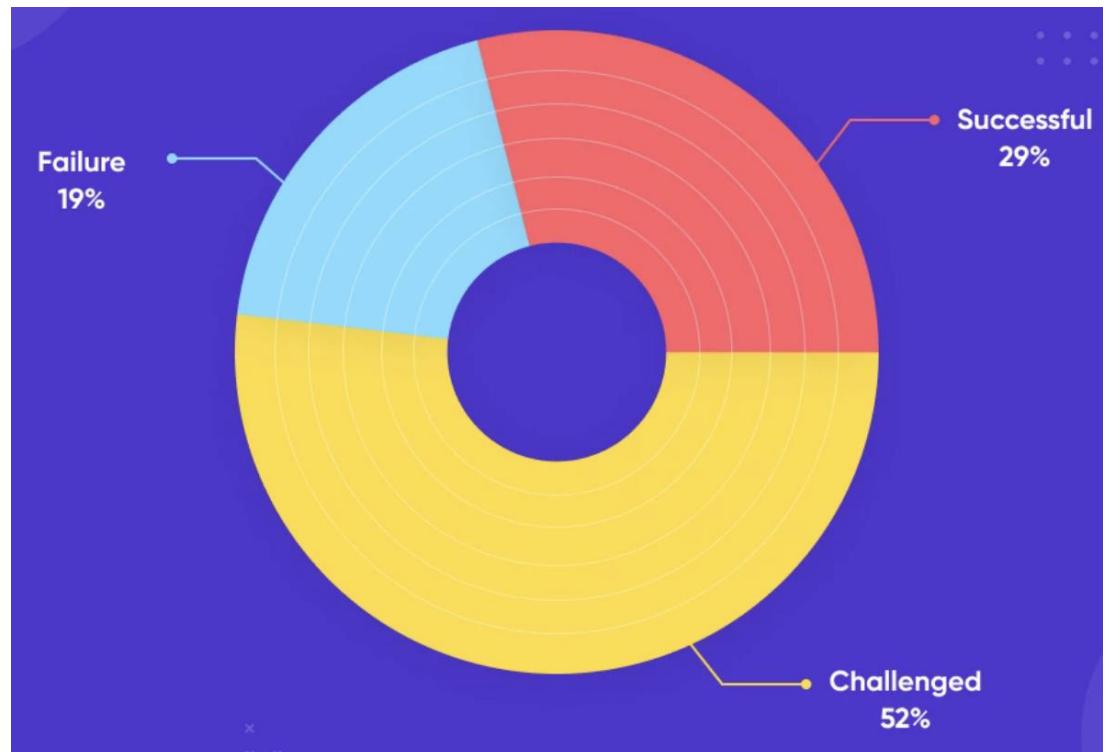
“Requirements are a specification of what should be implemented. They are descriptions of how the system should behave, or of a system property or attribute. They may be a constraint on the development process of the system.” [1]

[1] Ian Sommerville and Pete Sawyer (1997)

Success Rate of Software Projects

Did you know that **two-thirds** of all software projects end in partial or total failure?

The Standish Group discovered this when they analyzed **over 50,000** technology projects.



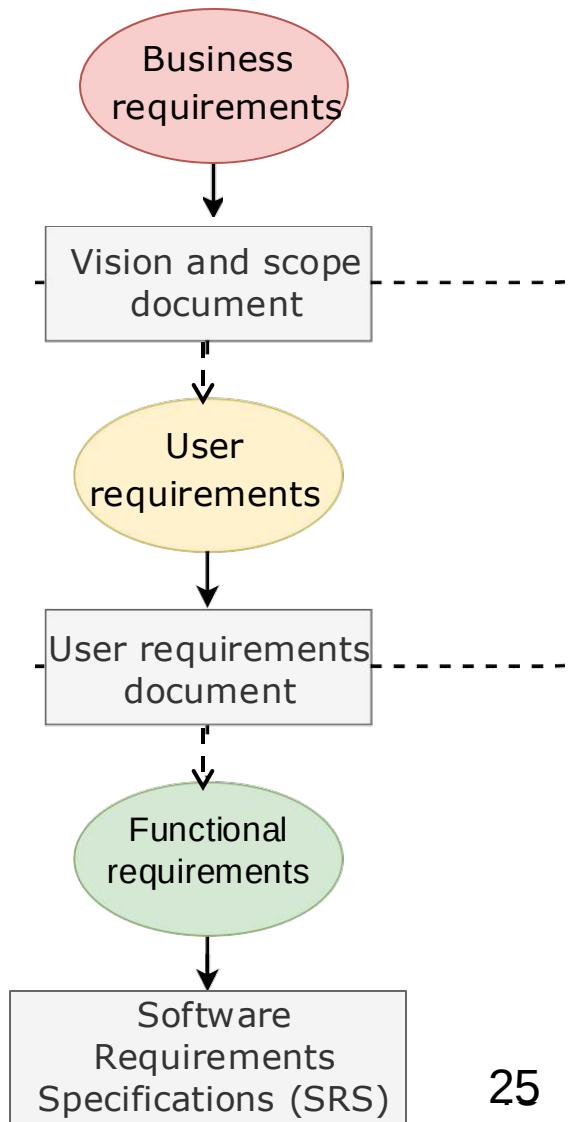
Objectives

- Why do we need to discover the requirements?
- What are the software requirements?
- **Levels and types of requirements**
- Product vs Project requirements
- Requirements engineering

Levels and types of requirements

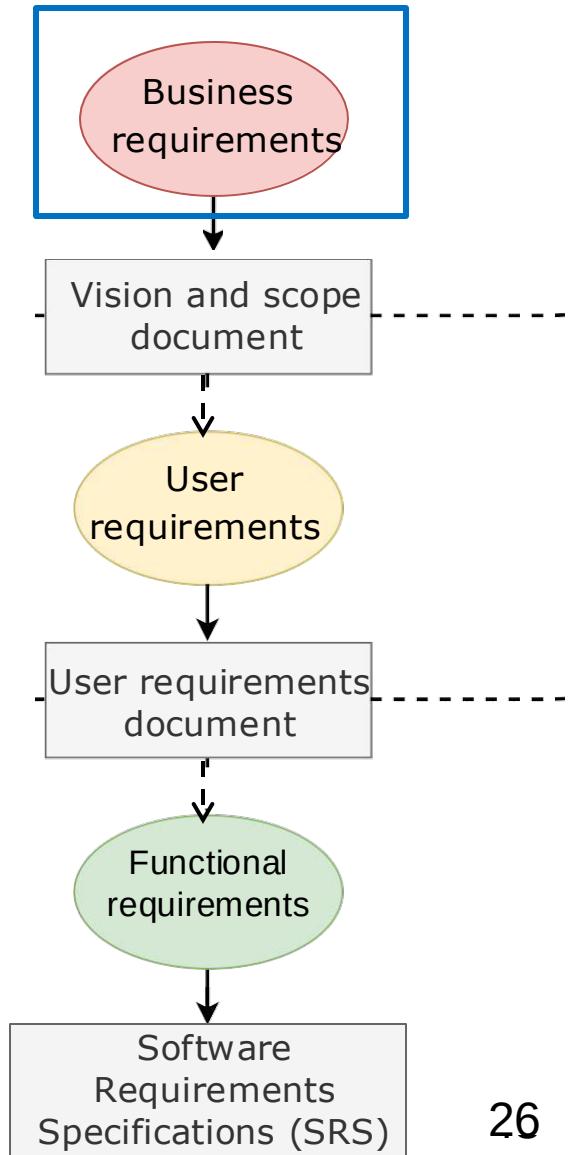
- We have three levels of **requirements**:
 - (1) business requirements
 - (2) user requirements
 - (3) functional requirements

Solid arrows mean “are stored in”; dotted arrows mean “are the origin of” or “influence.”



Levels and types of requirements

“Business requirements represent **high-level business objectives** of the **organization** that builds the product or of a **customer** who procures it.” [1]



How to identify the high-level business objectives?

- Business requirements reflect the answer to these questions:
 - 1) Why the organization is implementing this system?
 - 2) What are the benefits the organization hopes to achieve?

How to identify the high-level business objectives?



*“Suppose an airline wants **to reduce airport counter staff costs by 25%**. This goal might lead to the idea of building a kiosk that passengers can use to check-in for their flights at the airport.”* [1]

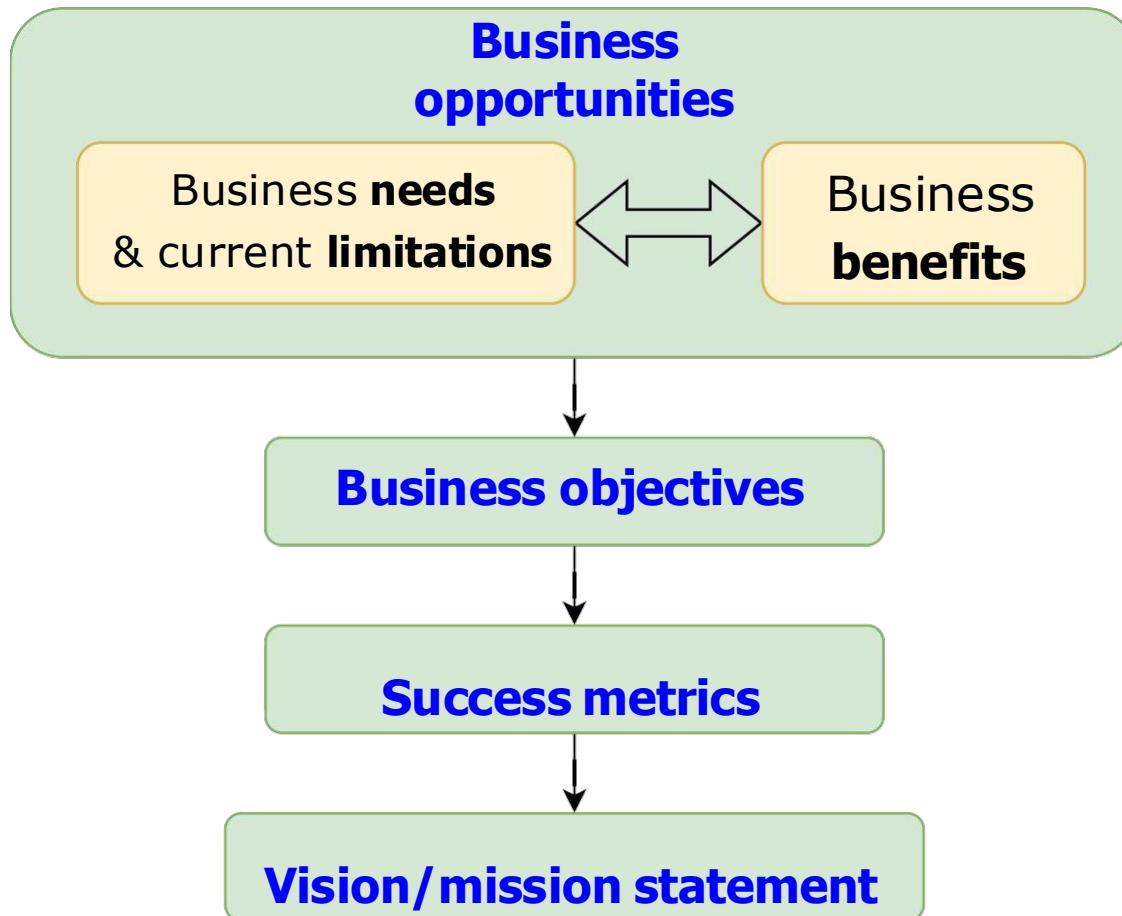
How to identify the high-level business objectives?



Implementing an online appointment scheduling system.

Business Objective?

How to identify the high-level business objectives?



How to identify the high-level business objectives?

- Business requirements
 - Defined by the business needs, market needs, or a new product concept
 - Stored in the “[vision and scope document](#)”
 - Include the [vision statement](#) or [mission statement](#)
 - Should represent short, simple, and clear objectives

How to identify the high-level business objectives?



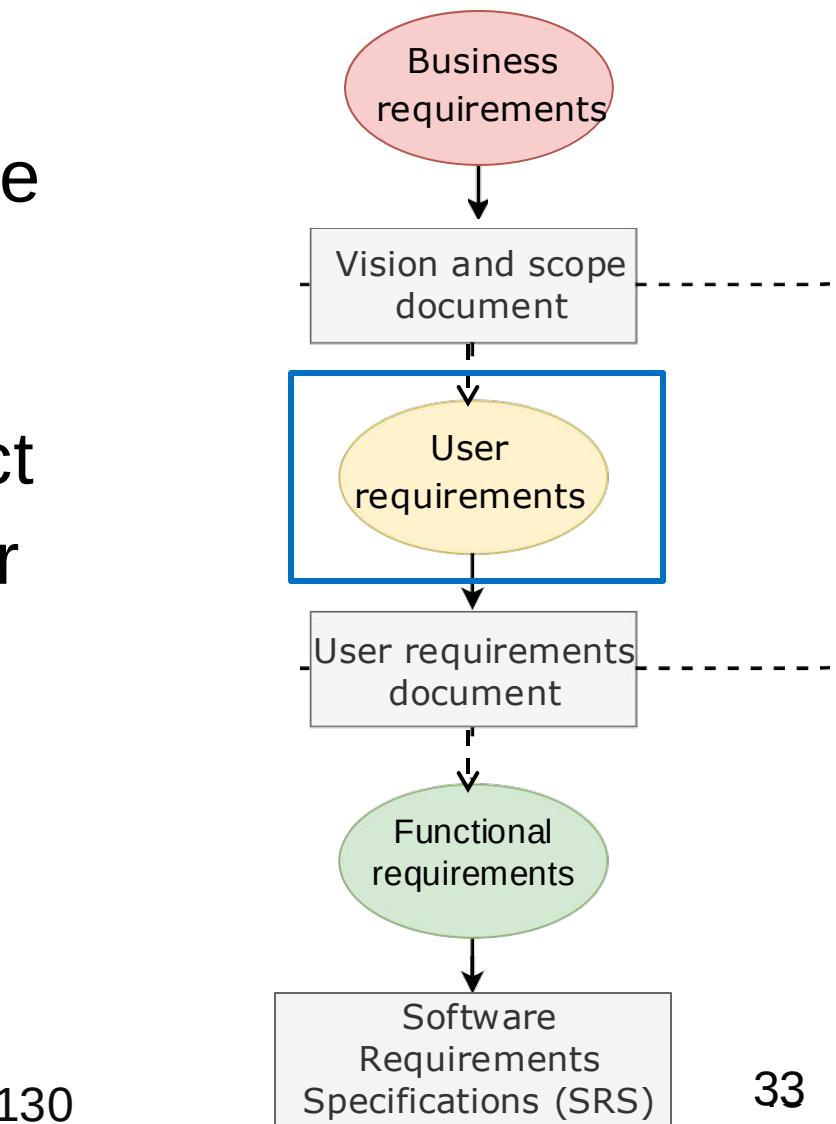
“Perhaps the most famous mission statement is President Kennedy’s ‘To put a man on the moon by 1970’” [1].

[1] Discovering requirements: How to specify products and services

Levels and types of requirements

User requirements describe

1. the **goals** or **tasks** the users must be able to perform with the product
2. the product **attributes** or **characteristics** that are important to user satisfaction.^[1]



Identifying user requirements

- User requirements should provide an answer to the question “**What the user will be able to do with the system?**”



For the airline system, “*Check-in for a flight*” and “*Check the flight status*” describe the airline system’s user requirements. [1]

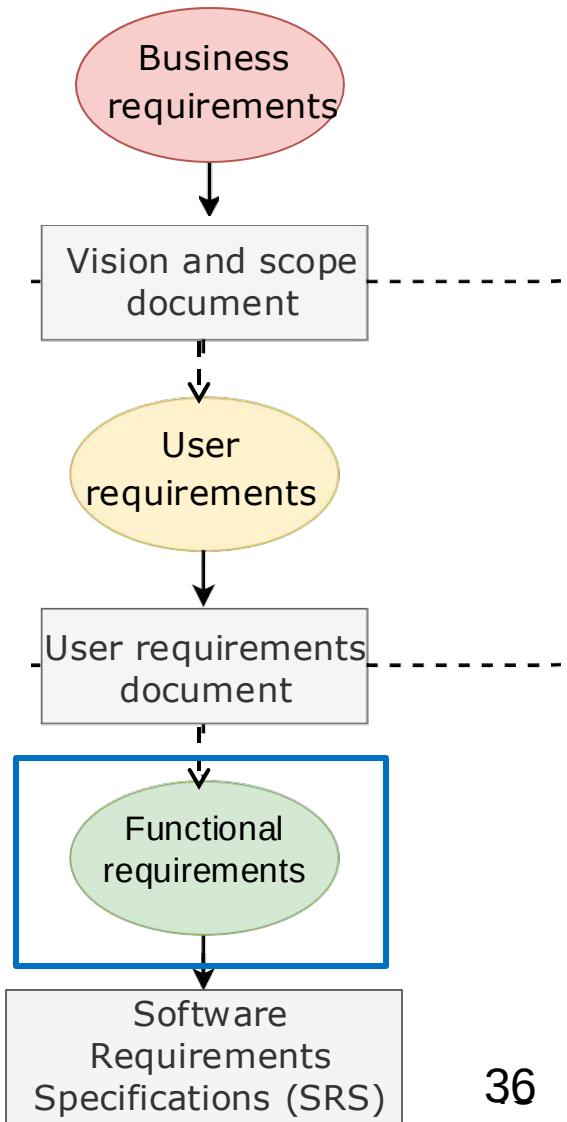
Identifying user requirements



For the healthcare system?

Levels and types of requirements

Functional requirements describe the behaviors the product will exhibit under specific conditions.



Identifying functional requirements

- Functional requirements should provide an answer to the question of what the developers must implement to enable users to accomplish their tasks (i.e., user requirements)
- Functional requirements are written using the “*shall statements*”

Identifying functional requirements



For the airline system,
“The Passenger shall be able to print boarding passes
for all flight segments for which she/he has checked in”

or “If the Passenger’s profile does not indicate a
seating preference, the reservation system shall assign
a seat.” [1]

Identifying functional requirements



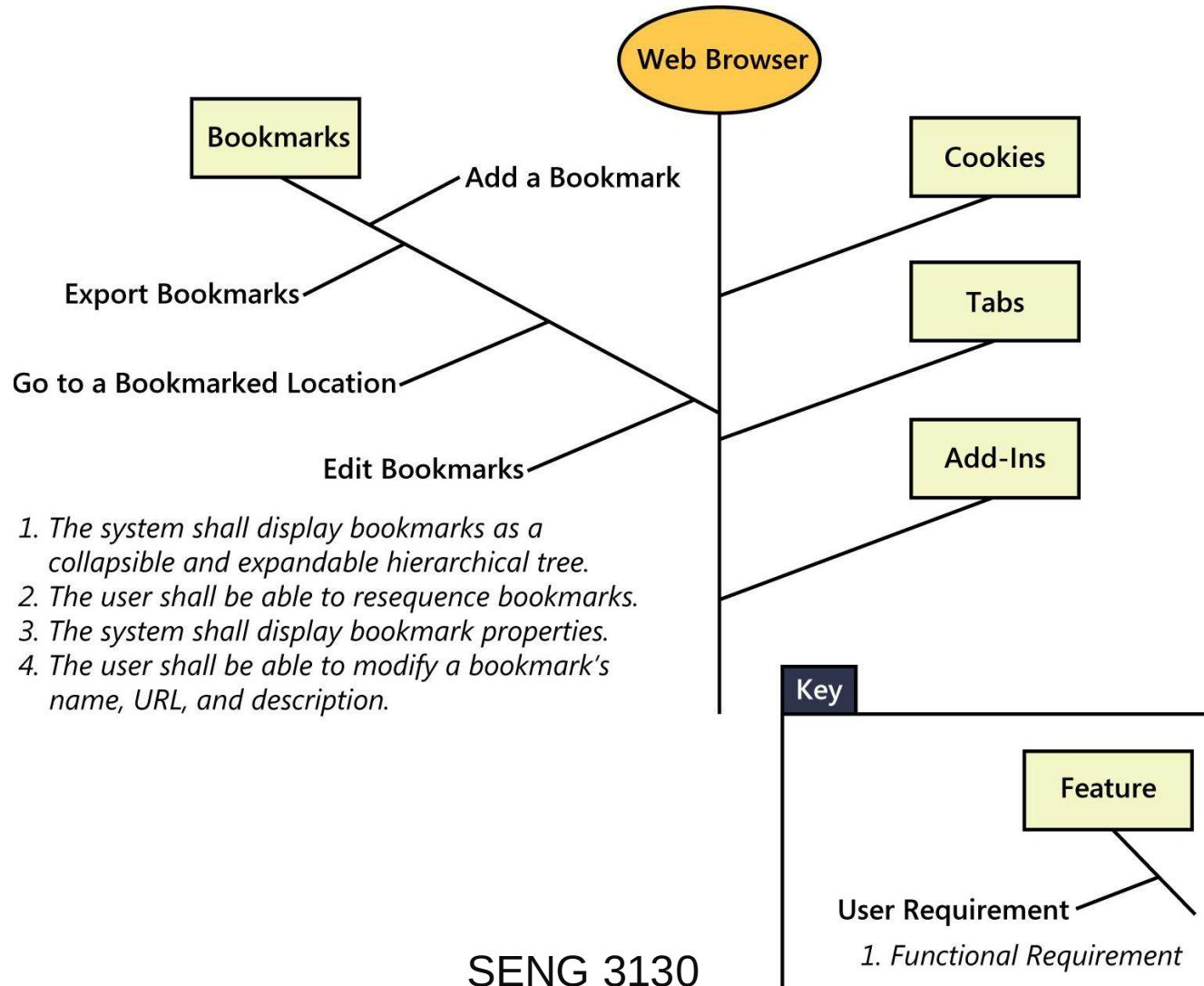
For the healthcare system?

Group user requirements into features

A *feature*:

- consists of one or more logically related system capabilities
- that provide value to a user
- and are described by a set of functional requirements

Group user requirements into features

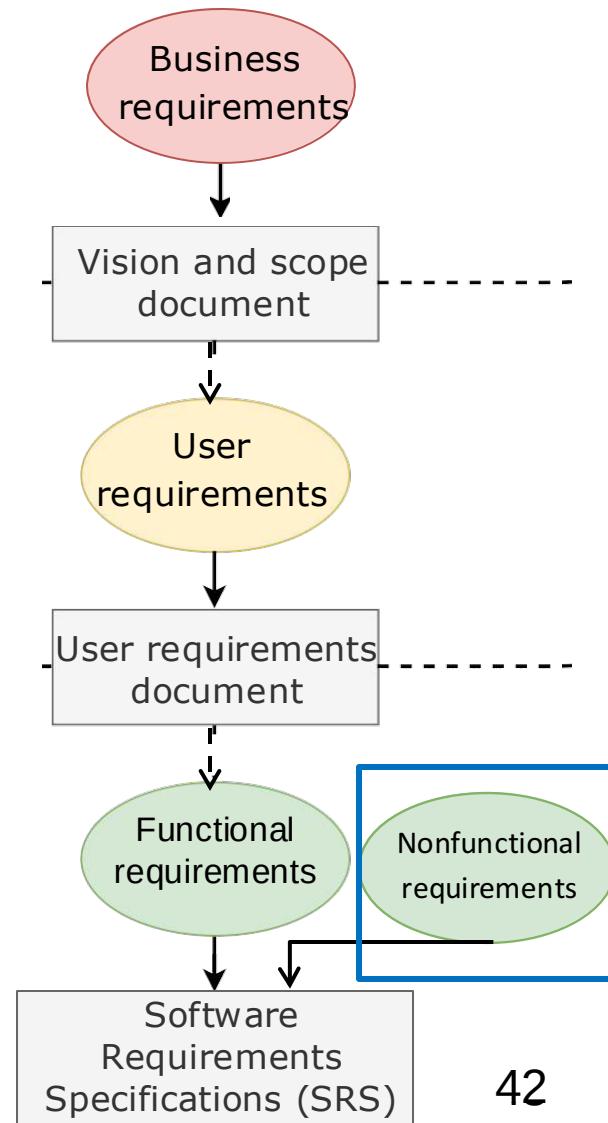
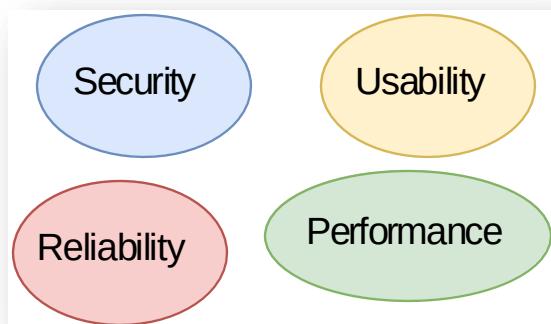


What are the non-functional requirements?

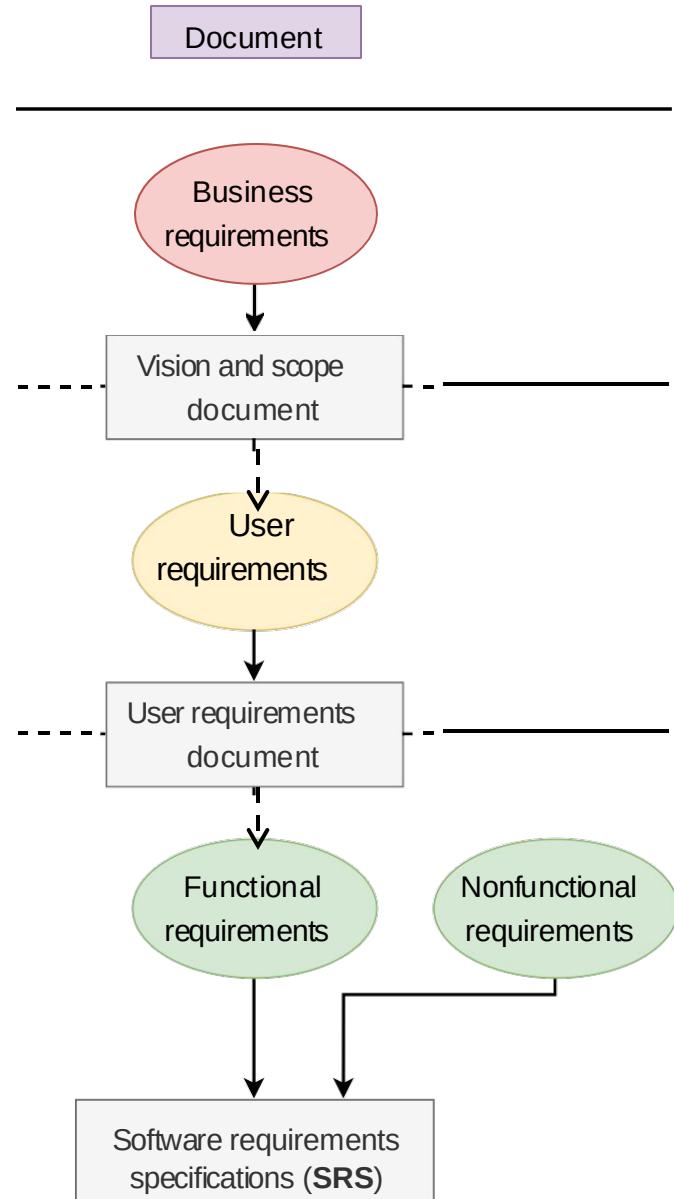
Non-functional requirements describe

- 1) quality attributes
- 2) constraints
- 3) external interfaces of the product

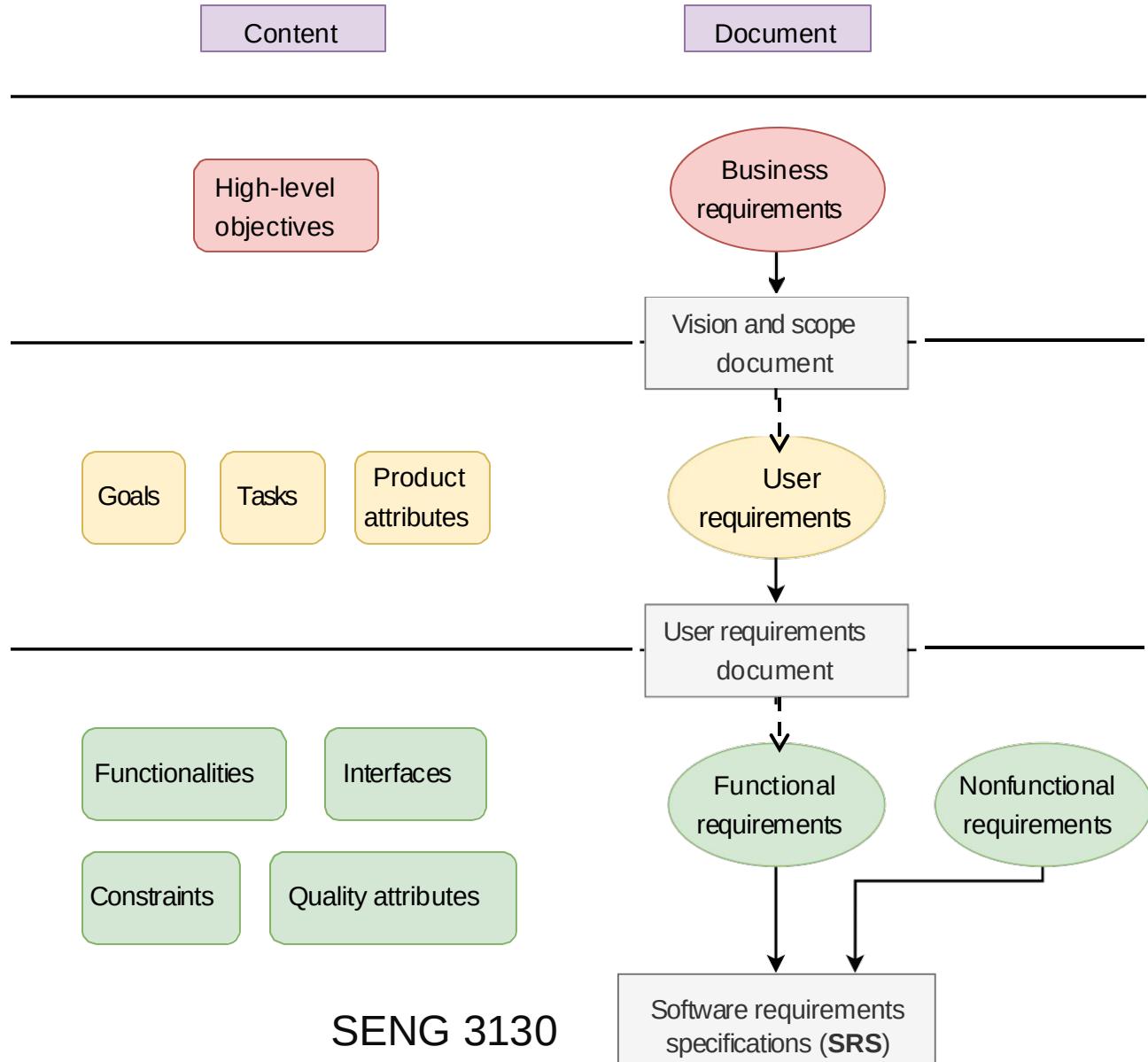
Other-than-functional requirements might specify not *what* the system does, but rather *how well* it does those things.



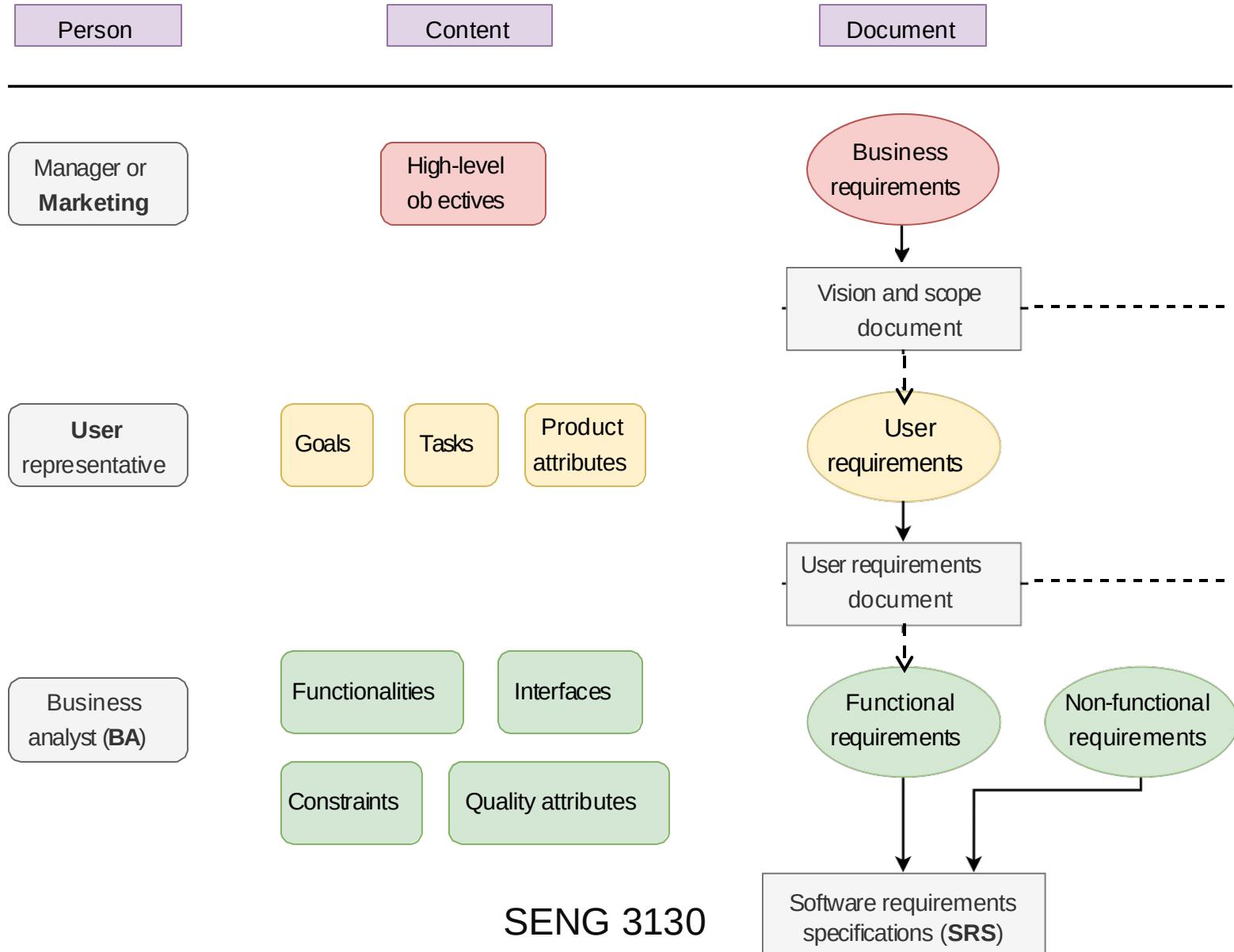
Connecting all pieces



Connecting all pieces



Connecting all pieces



Objectives

- Why do we need to discover the requirements?
- What are the software requirements?
- Levels and types of requirements
- **Product vs. Project requirements**
- Requirements engineering

Product vs. Project requirements

- **Product requirements** represent business requirements, user requirements, functional and non-functional requirements.
- **Project requirements** represent the resources needed to plan, develop, deploy, and maintain the project, such as **training** and **infrastructure capabilities**.

Requirements of the “Online Purchase System”

Auditing manager can export a summary report of the purchases of the last 12 months

Business requirements

Users shall enter his name and password to login

User requirements

Buyers need a system that speeds up the ordering process

Functional requirements

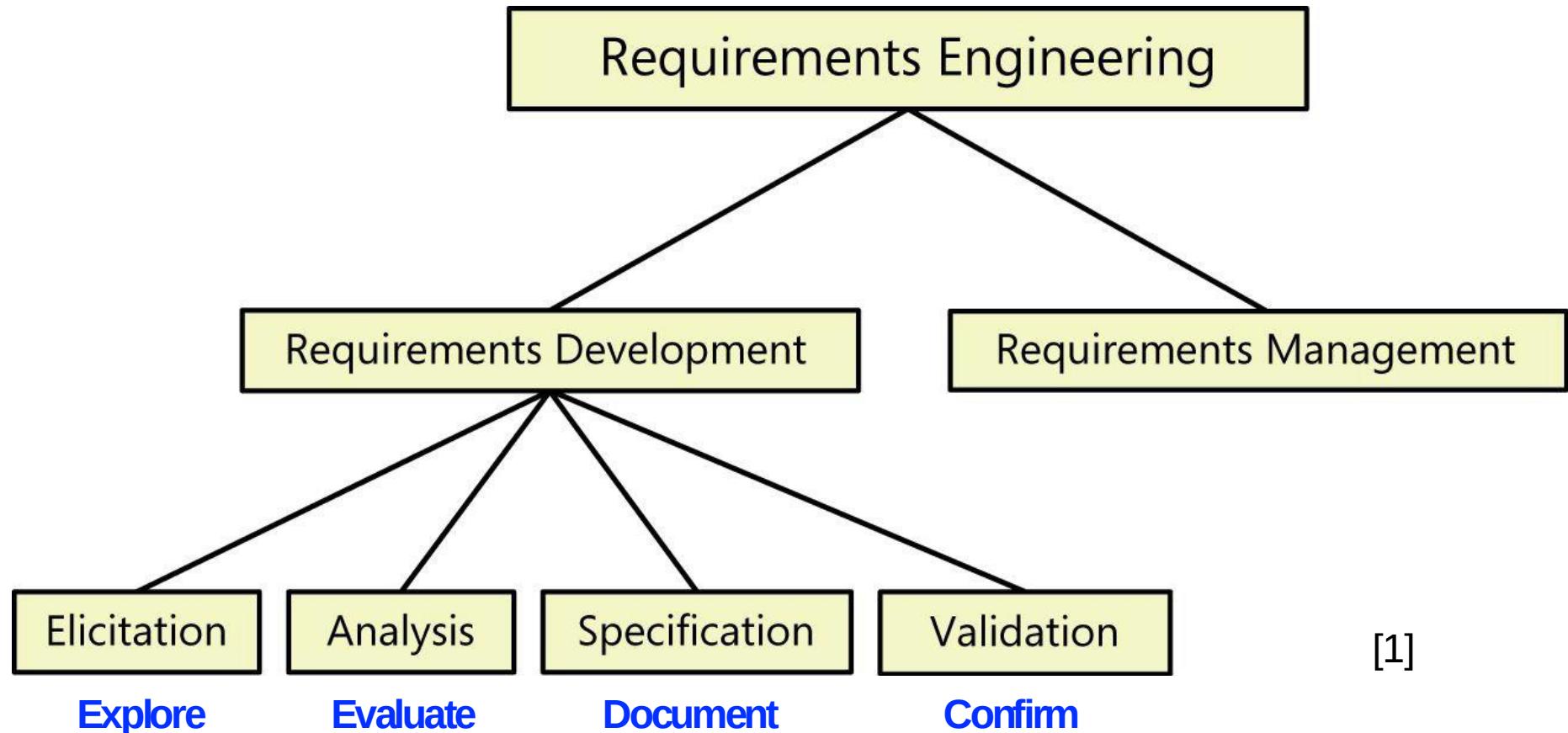
The system needs to be deployed on 60 different machines distributed across two continents

Project requirements

Objectives

- Why do we need to discover the requirements?
- What are the software requirements?
- Levels and types of requirements
- Product vs. Project requirements
- **Requirements engineering**

Requirements Engineering (RE)



Requirements Engineering (RE)

Elicitation **Explore**

interviews, workshops, document analysis, prototyping

- Identifying the product's **expected user classes** and other stakeholders.
- Understanding user **tasks** and **goals** and the **business objectives**
- Learning about the **environment** in which the new product will be used.
- Working with **individuals** who represent each user class

Requirements Engineering (RE)

Analysis Evaluate

reaching a richer and more precise understanding of each requirement and representing sets of requirements in multiple ways.

Identifying gaps in requirements or unnecessary requirements as they relate to the defined scope

Requirements Engineering (RE)

Specification **Document**

Translating the collected user needs into **written** requirements and **diagrams** suitable for **comprehension, review**, and **use** by their intended audiences.

Requirements Engineering (RE)

Validation **Confirm**

The **correct** set of requirements information
that will enable **developers** to build a solution
that **satisfies** the business objectives.

Requirements Engineering (RE)

Requirements Management

- Evaluating the impact of proposed requirements **changes**
- Incorporating **approved** changes into the project in a controlled way
- Keeping **project plans** current with the requirements as they evolve
- Defining the **relationships and dependencies** that exist between requirements
- Tracing individual requirements to their corresponding **designs**, **source code**, and **tests**

Summary

- Identifying requirements-related issues in the **early phases** of the project saves money, effort, and company reputation.
- A business analyst (BA) explores requirements in phases.
 - Each phase describes the requirements from a **particular perspective** and with a certain **level of detail**.
- RE is an **iterative process** through which the requirements are explored, evaluated, documented, and confirmed.

Things to do

- Read chapter 1 of the textbook
- Start building your team for the course project

References

1. Software Requirements (Developer Best Practices)
Karl Wiegers, Joy Beatty, 3rd Edition, Microsoft Press,
2013, ISBN-10: 0735679665

2. Software Requirements Essentials: Core Practices for
Successful Business Analysis Karl Wiegers and
Candase Hokanson, 1st Edition, Addison-Wesley
Professional, 2023.