CS304 SOFTWARE ENGINEERING

Yida Tao

taoyd@sustech.edu.cn

Yida Tao

taoyd@sustech.edu.cn

WHERE ARE WE NOW?

Requirement analysis
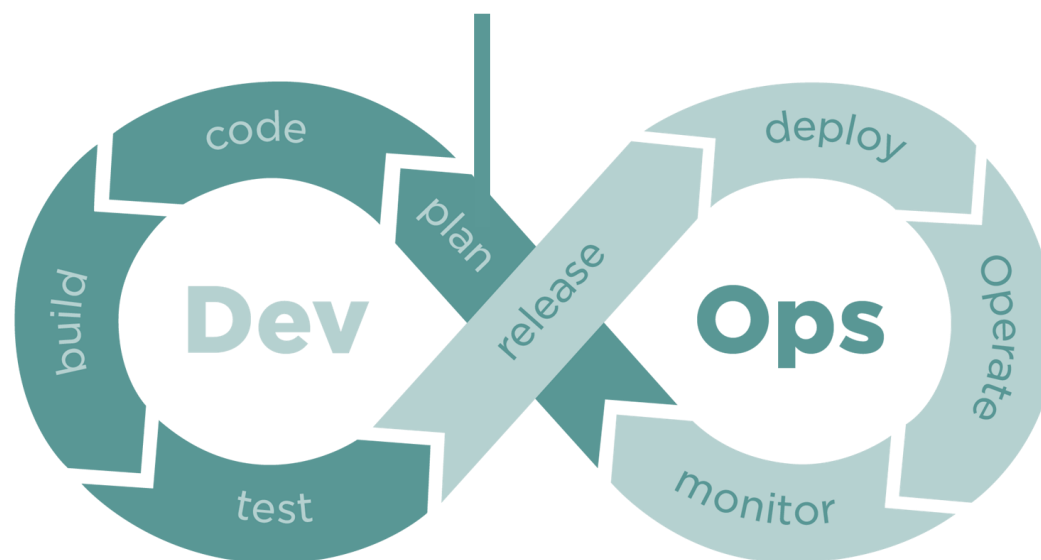
TAO Yida@SUSTECH

LECTURE 4

- **Overview of software requirements**

- **Stakeholders**

- **Types of requirements**

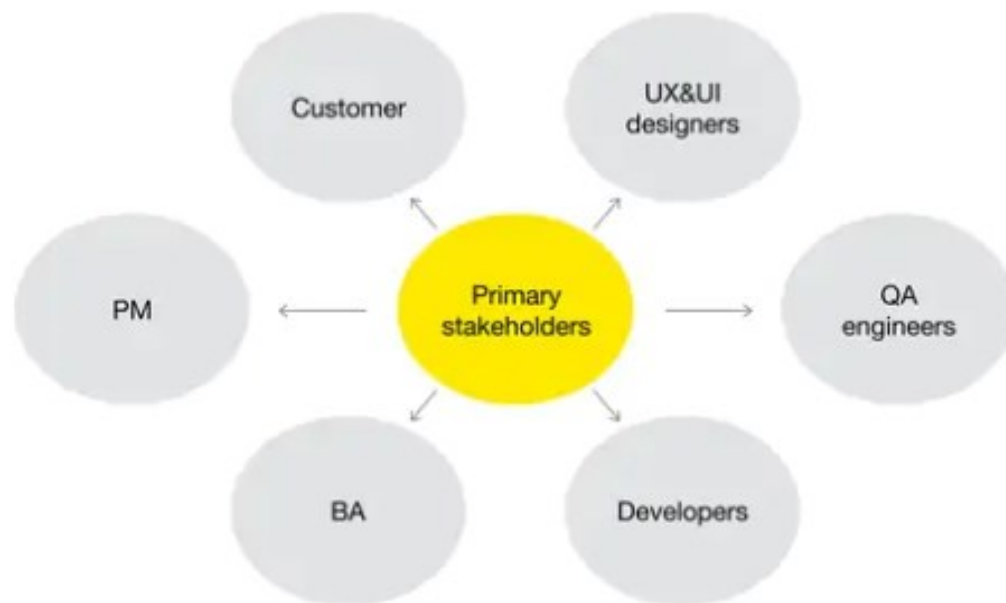- **Requirements analysis in Scrum**

WHAT IS SOFTWARE REQUIREMENT?

- Software requirements (需求) describe the following:
  - what the system should do
  - the services that it provides
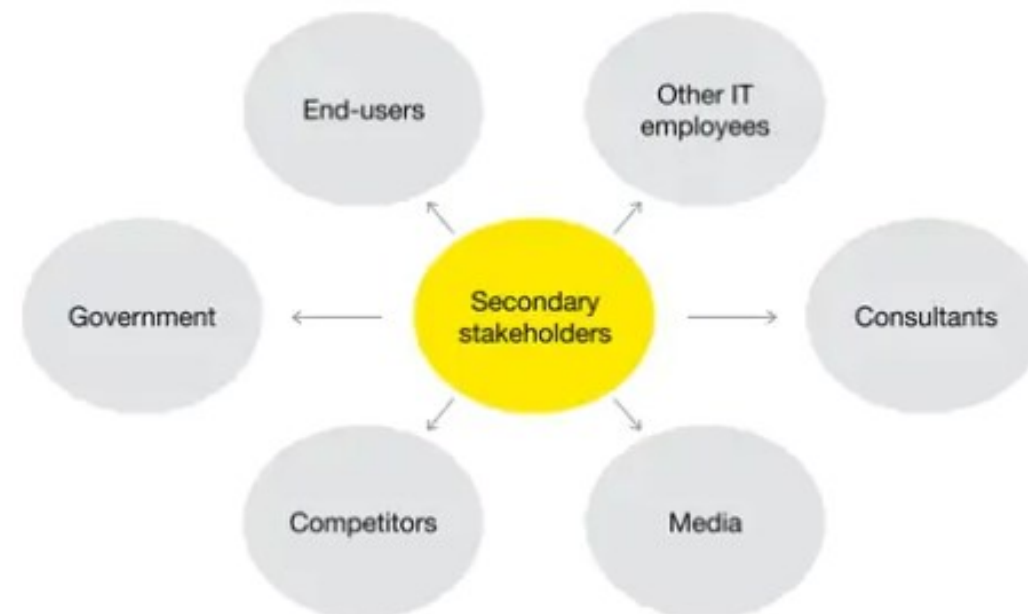 - the constraints on its operation

Who decides these requirements?

STAKEHOLDERS

- 利益相关者: Anyone who affects or is affected by the software.

Primary stakeholders have a direct impact on your software project. They are people, groups, or organizations that have the strongest voice and can gain or lose their income.

Secondary stakeholders have an indirect relationship with a software development process. They do not have any direct engagement with a project or a company but can indirectly affect decisions related to your software product and its reputation.
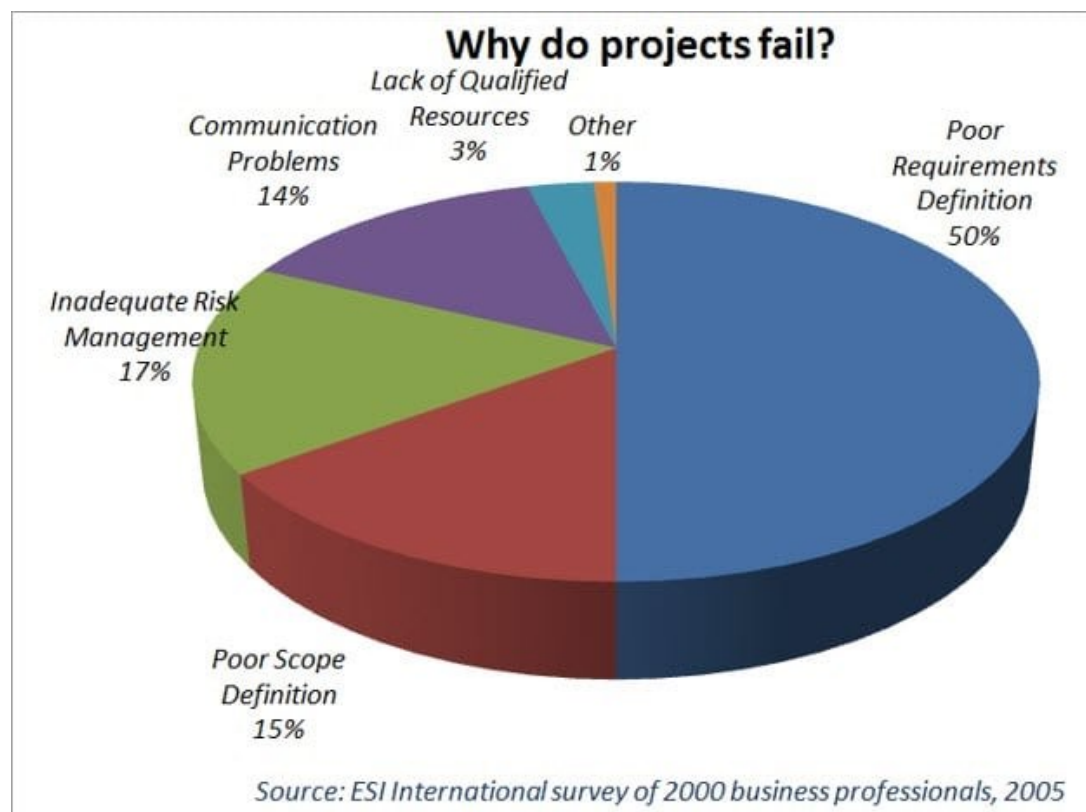
STAKEHOLDERS

Ignoring stakeholders in requirement analysis might lead to project failure

Case study

- In 2010, Microsoft launched the Kin phone for young users.
- The product **failed to meet end-user needs** (e.g., too expensive for teenagers, no App store)
- Poor sales. Died in just 2 months!

WHY ARE REQUIREMENTS IMPORTANT?

Why do projects fail?

- Lack of Qualified Resources 3%
- Communication Problems 14%
- Other 1%
- Poor Requirements Definition 50%
- Inadequate Risk Management 17%
- Poor Scope Definition 15%

Source: ESI International survey of 2000 business professionals, 2005

WHY ARE REQUIREMENTS IMPORTANT?

Figure 1-2. *Timeline of the developer workflow*
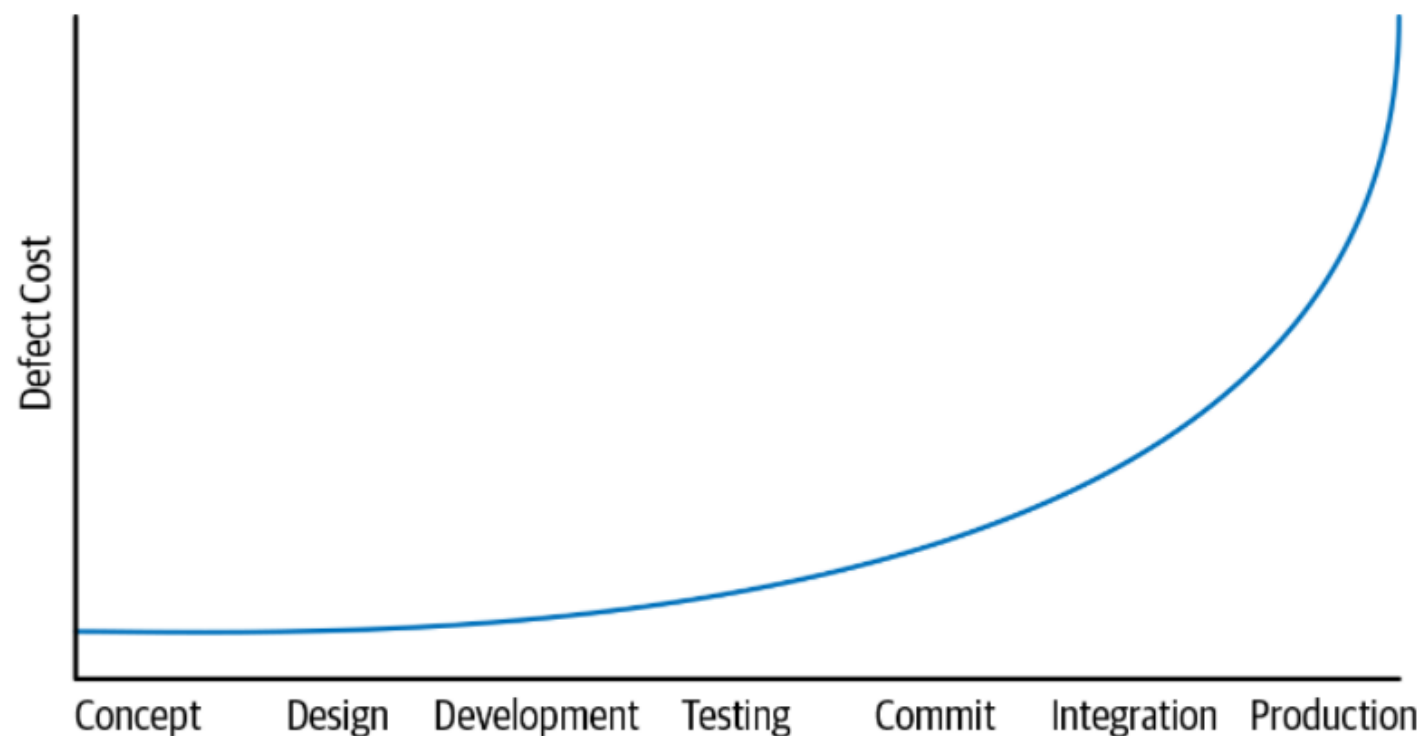
The later you identify requirement deficiencies, the higher the cost of resolving them

WHY ARE REQUIREMENTS IMPORTANT?

50% of source code defects can be attributed to deficiencies in requirements
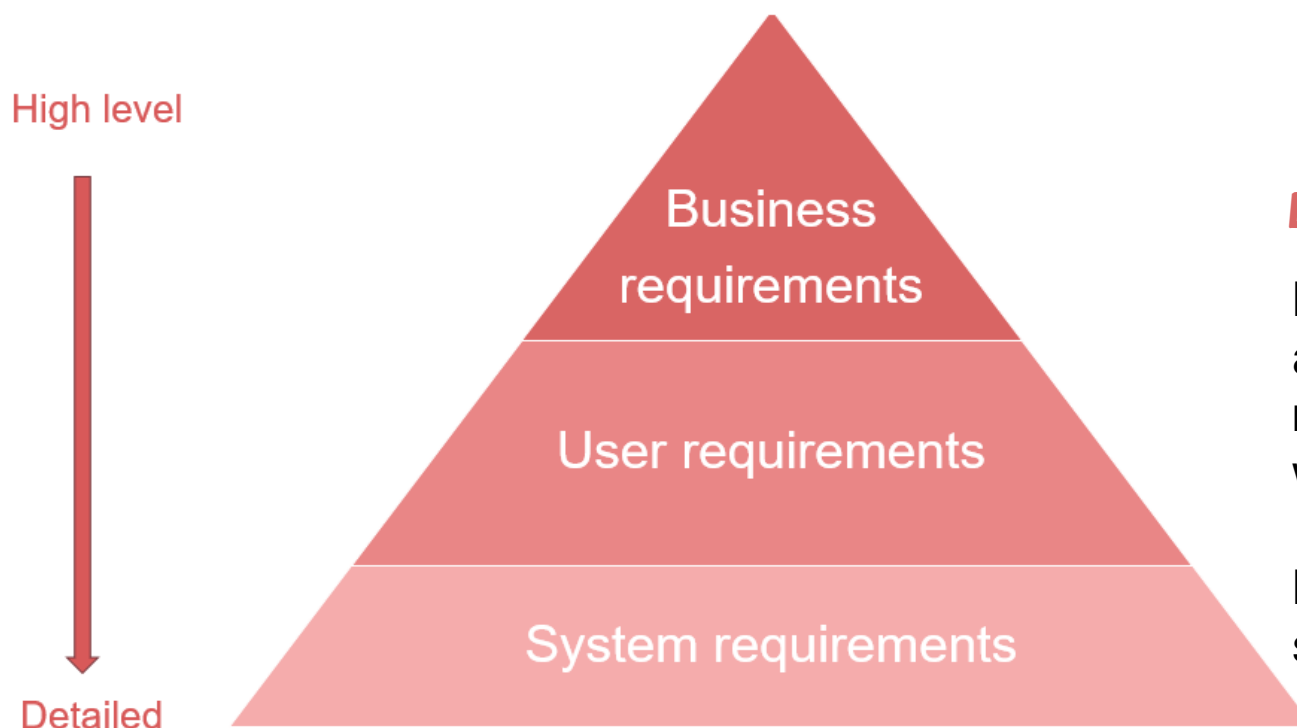
Source: 《现代软件工程基础》

LECTURE 4

- Overview of software requirements
- Stakeholders
- **Types of requirements**
- Requirements analysis in Scrum

CLASSIFICATION OF SOFTWARE REQUIREMENTS

High level

Detailed

Business requirements
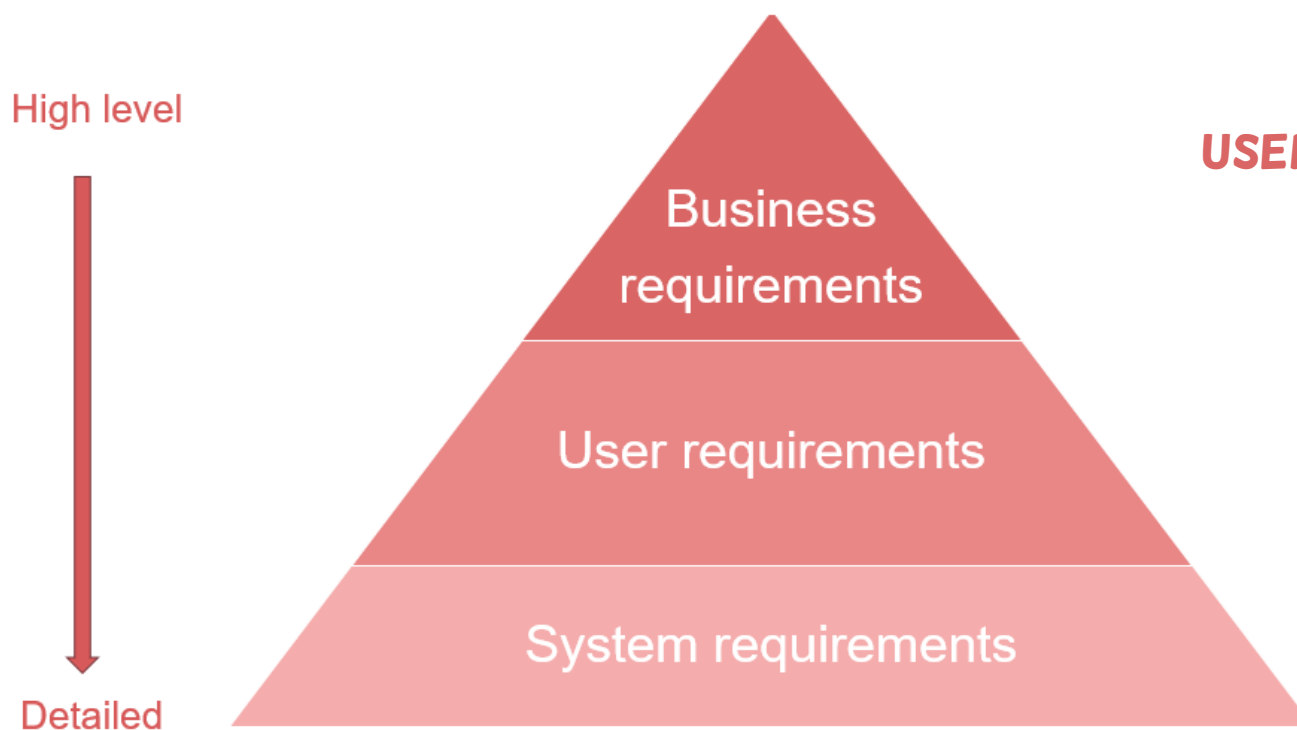
User requirements

System requirements

BUSINESS VIEW: WHY IS THE PROJECT NEEDED?

Business Requirements outline a general overview of a product, such as its primary usage, why it is needed, its scope and vision, what business benefits will be gain, intended audience or users, etc.

Business Requirements define the **WHY** behind a software project.

CLASSIFICATION OF SOFTWARE REQUIREMENTS

High level

Detailed

Business requirements

User requirements

System requirements

USER VIEW: WHAT DO USERS NEED THE SYSTEM TO DO?

User requirements are gathered using use case, user scenarios, and user stories

Often written in natural languages with **little/no technical details**

User requirements describe the **WHO** of a software project.

CLASSIFICATION OF SOFTWARE REQUIREMENTS

High level

Detailed

Business requirements

User requirements

System requirements
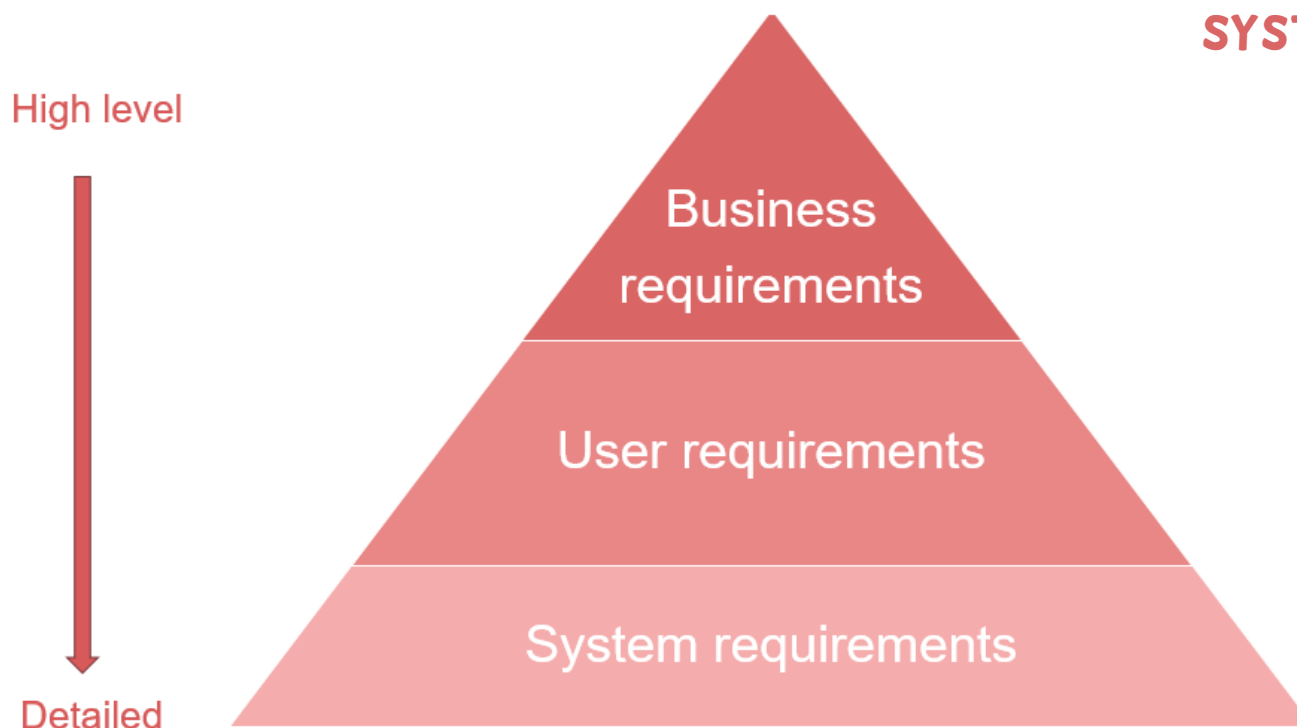
SYSTEM VIEW: WHAT DOES THE SYSTEM NEED TO DO?

System requirements describe software as **functional** modules and **non-functional** attributes.
- Functional requirements
- Non-functional requirements

System requirements are written for **developers** with more formal format

System requirements dive into the **HOW** of a software project.

EXAMPLE

Business requirements, as visions (愿景), need to be iteratively decomposed (分解) and refined (精化), until achieving detailed and actionable system requirements

DECOMPOSED AND REFINED

VISION
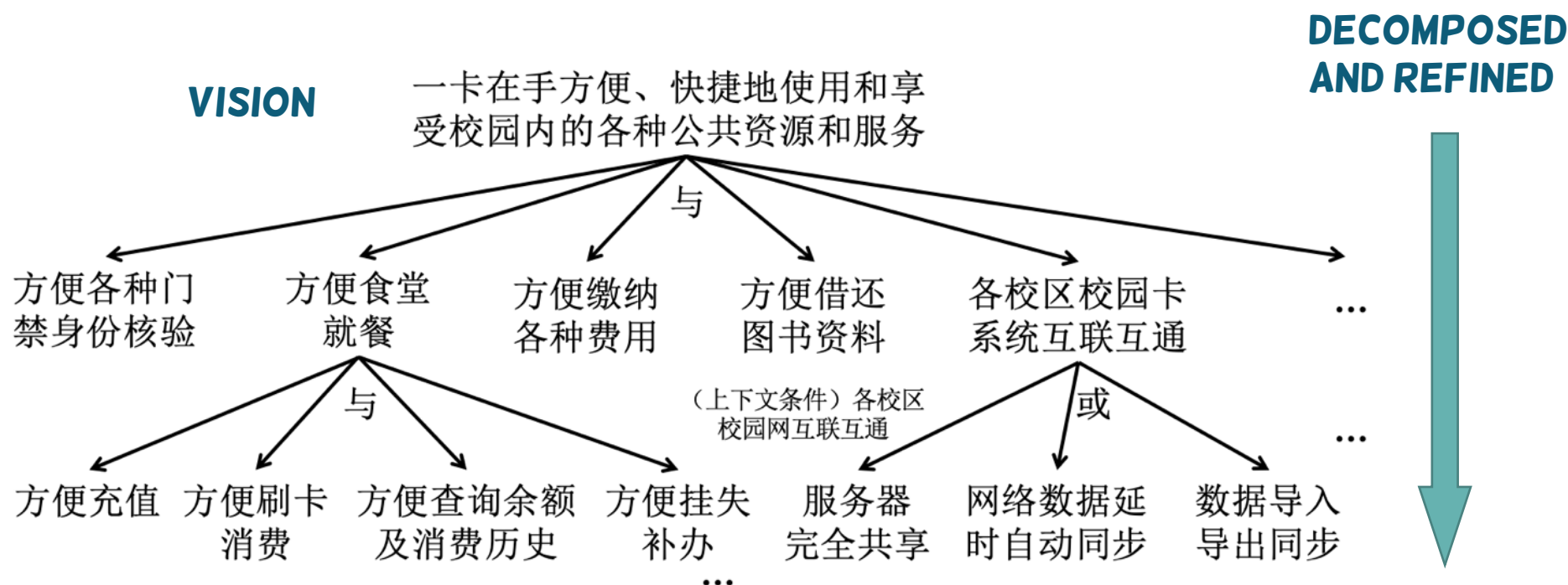
FUNCTIONAL REQUIREMENTS

- Functional requirements are basic functionalities that the system should offer.

- They are represented or stated in the form of **input** to be given to the system, the **operation** performed and the **output** expected.

- Functional requirements are basically the requirements stated by the user which one can **see directly in the final product**

FUNCTIONAL REQUIREMENTS

Subtypes	Examples
系统应提供的服务 (service provided)	能够按照关键字检索图书 (searching books by keywords)
系统针对特定输入的响应 (actions for specific input)	对于格式不正确的身份证号进行提示并请用户重新输入 (re-enter ID for invalid input)
系统在特定情形下的行为 (behavior for specific context)	用户如果5分钟内没有操作，那么主界面自动进入锁定状态 (lock the screen if no user operations for 5 min)
系统不应做什么 (forbidden activity)	不允许尝试密码输入三次以上 (cannot input password over 3 attempts)

NON-FUNCTIONAL REQUIREMENTS

- Non-functional requirements are not related to the software's functional aspect

- Non-functional requirements specify the software's **quality attribute**.

- Non-functional requirements define the general characteristics, behavior of the system, and features that **affect the experience of the user**.

NON-FUNCTIONAL REQUIREMENTS

Subtypes	Examples
性能（performance）	联机刷卡应当在5秒内返回结果 (Online card swiping should return results within 5 seconds.)
可靠性（reliability）	系统的整体可靠性要达到99.99%以上
安全性（security）	系统应确保手机支付充值账户和密码不会被泄漏和盗用 (payment accounts and passwords will not be leaked or misused.)
易用性（usability）	用户根据提示学会手机支付充值的时间不超过10分钟 (Users should learn mobile payment within 10 minutes)
产品约束 (product constraints)	软件系统要在已有的几台服务器上运行并使用Linux系统 (Software needs to run on existing Linux servers)
过程约束 (process constraints)	软件系统应当在5个月内交付并严格遵循给定的过程规范 (Software should be delivered in 5 month)

LECTURE 4

- **Overview of software requirements**

- **Stakeholders**

- **Types of requirements**

- **Requirements analysis in Scrum**

REQUIREMENT ANALYSIS IN SCRUM

- The product owner is responsible for talking to all stakeholders and gathering requirements.

- Based on the gathered information, the scrum team work together to come up with user stories, which are simple sentences that describe what the users need

COMMON FORMAT OF USER STORY

As a «role/profile»

I want to «action/activity»

so that «benefit/reason»

USER STORY EXAMPLES

As a
registered user

I want to
change my password

So I can
keep my account secure

As a
website visitor

I want to
susbscribe to the mailing list for a product

So I can
get product updates through email

https://www.youtube.com/watch?v=apOvF9NVguA

USER STORY EXAMPLES

As an
admin user
I want to
disable a user
So I can
prevent unauthorized logins by past
employees

As a
mobile app user
I want to
save all my data to the cloud
So I can
access it from another device

https://www.youtube.com/watch?v=apOvF9NVguA

MORE EXAMPLES

- As a participant, I want to register for an event so that I can attend it.

- As a sales professional, I want to generate reports so that I can take a decision on the marketing strategy for the upcoming quarter

- As a restaurant owner, I want to update my menu items in real-time on my website, so that my customers have accurate information.

- As an avid reader, I want to receive personalized book recommendations based on my reading history, so that I can discover new books that align with my interests.
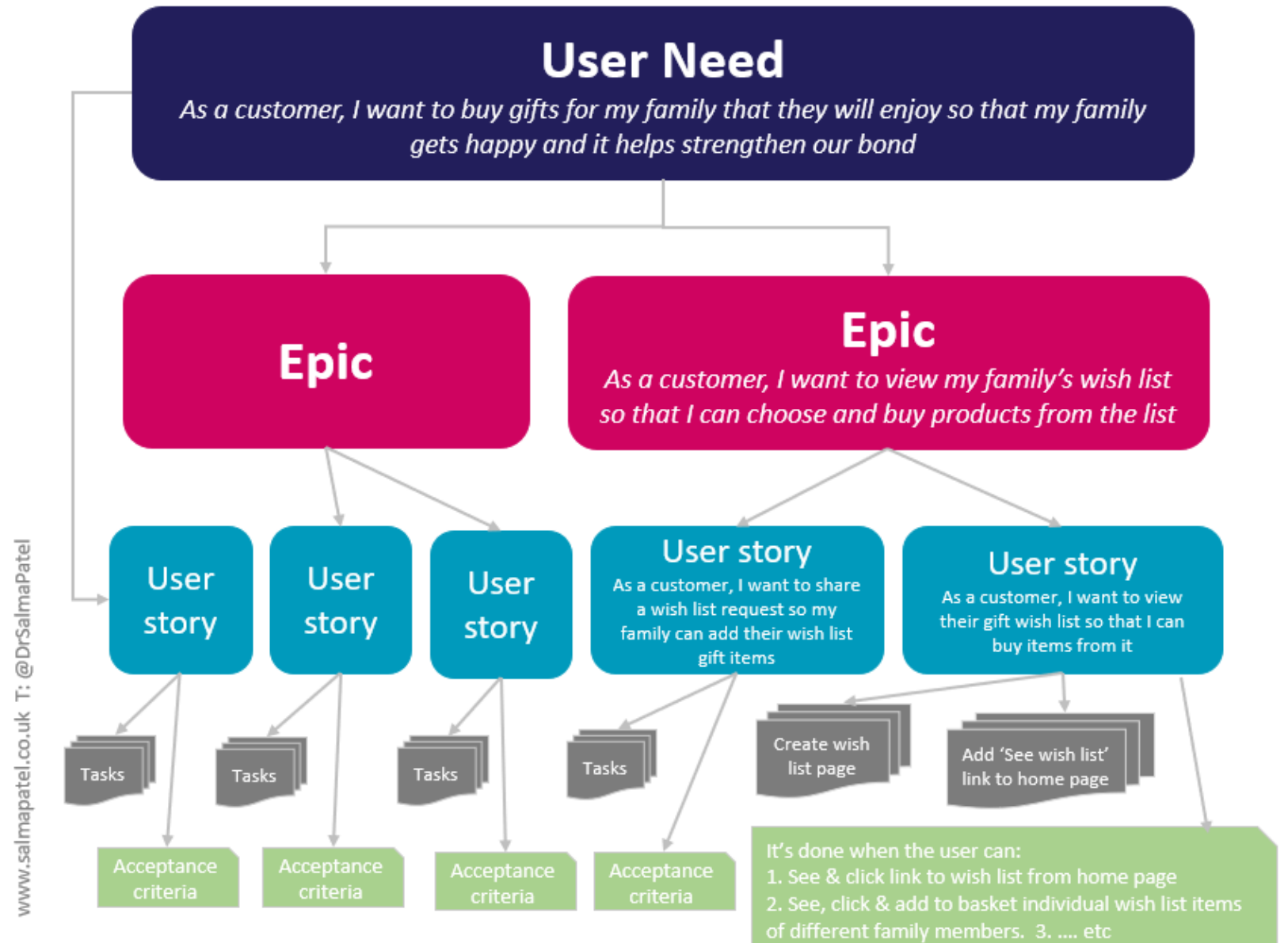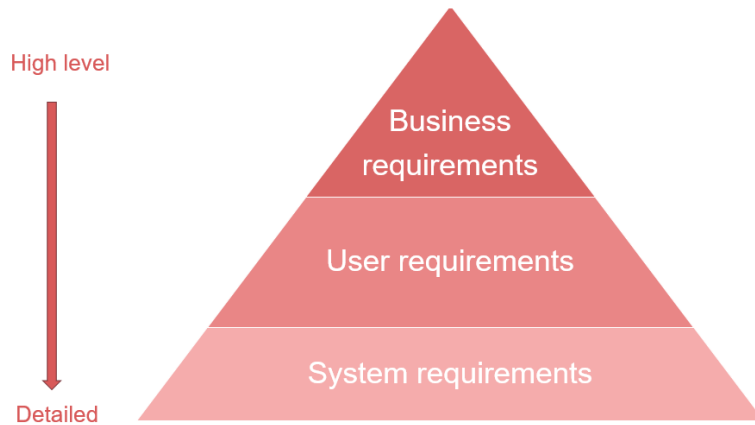
WHAT MAKES A GOOD USER STORY?

- Stories come from the perspective of **users**
- Stories create **business value** for the customers

Bad user story:
"As a developer, I want a database with all the tables to model the data, so I can store information the application needs"

GRANULARITY OF USER STORY

High level

Business requirements

User requirements

System requirements

Detailed

User Need
As a customer, I want to buy gifts for my family that they will enjoy so that my family gets happy and it helps strengthen our bond

Epic

Epic
As a customer, I want to view my family's wish list so that I can choose and buy products from the list

User story

User story

User story

User story
As a customer, I want to share a wish list request so my family can add their wish list gift items

User story
As a customer, I want to view their gift wish list so that I can buy items from it

Tasks

Tasks

Tasks

Tasks

Create wish list page

Add 'See wish list' link to home page

Acceptance criteria

Acceptance criteria

Acceptance criteria

Acceptance criteria

It's done when the user can:
1. See & click link to wish list from home page
2. See, click & add to basket individual wish list items of different family members. 3. .... etc

www.salmapatel.co.uk T: @DrSalmaPatel
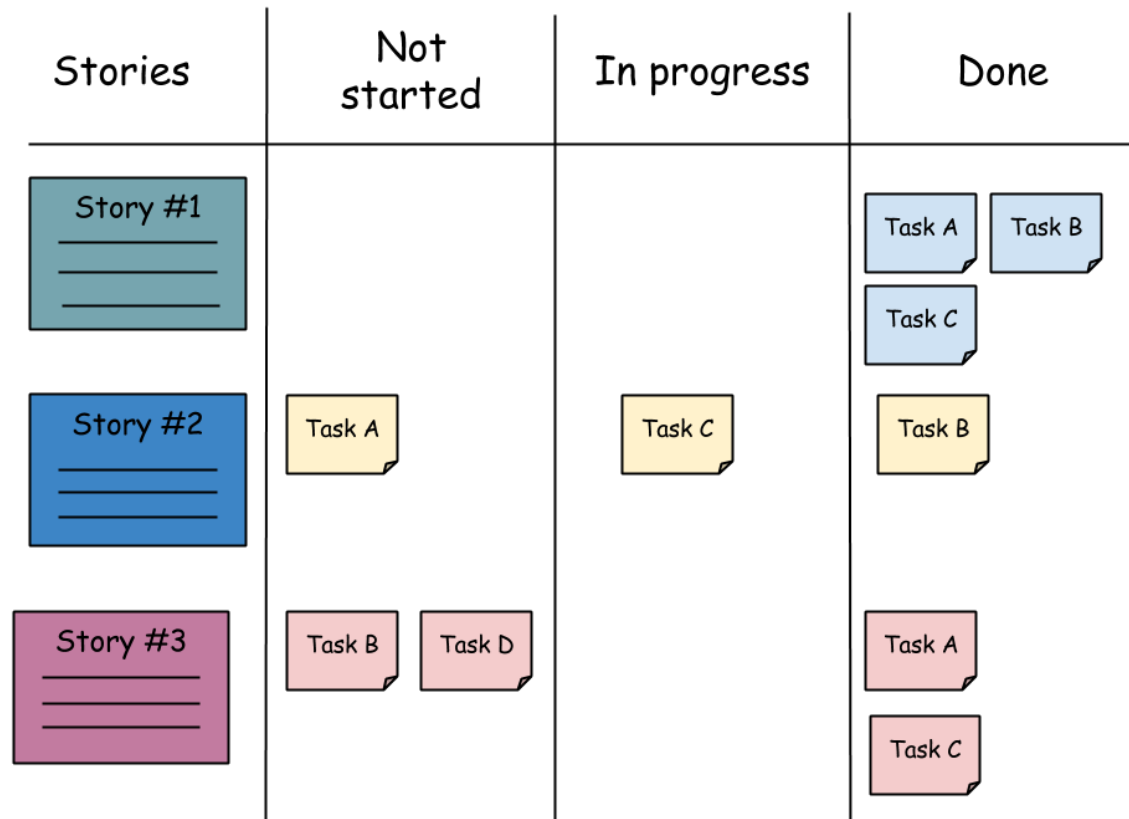
FROM USER STORIES TO TASKS

- Tasks are used to break down user stories even further.

- Tasks are the smallest unit used in scrum to track work.

- A task should be completed by one person on the team

- A task typically takes ~1 day

FROM USER STORIES TO TASKS

As a customer, I want to be able to have wishlists so that I can come back to buy products later

User story

| Put 'Add to wishlist' button on each product page | Create new db to store wishlist items | Create page to display user's wishlist | Add 'View wishlist' link to homepage |

Tasks

Tasks should be small, manageable, and clear, so that everyone knows what need to be done.

FROM USER STORIES TO TASKS

HOW TO PRIORITIZE USER STORIES?

MoSCoW Model

M MUST HAVE
Non-negotiable product needs that are mandatory for the team

S SHOULD HAVE
Important initiatives that are not vital, but add significant value.

C COULD HAVE
Nice to have initiatives that will have a small impact if left out.

W WILL NOT HAVE
Initiatives that are not priority for this specific time frame.

As a student, I want to submit assignments online so that I can complete my coursework.

As an instructor, I want to grade assignments and provide feedback so that students can improve their performance.

As a student, I want to receive push notifications for assignment deadlines so that I never miss a due date.

As a student, I want to personalize my dashboard (theme, widgets) so that I can improve my learning experience.

As an admin, I want to support blockchain-based certificates for students so that credentials are more secure and verifiable.

M — **MUST HAVE**
Non-negotiable product needs that are mandatory for the team

S — **SHOULD HAVE**
Important initiatives that are not vital, but add significant value.

C — **COULD HAVE**
Nice to have initiatives that will have a small impact if left out.
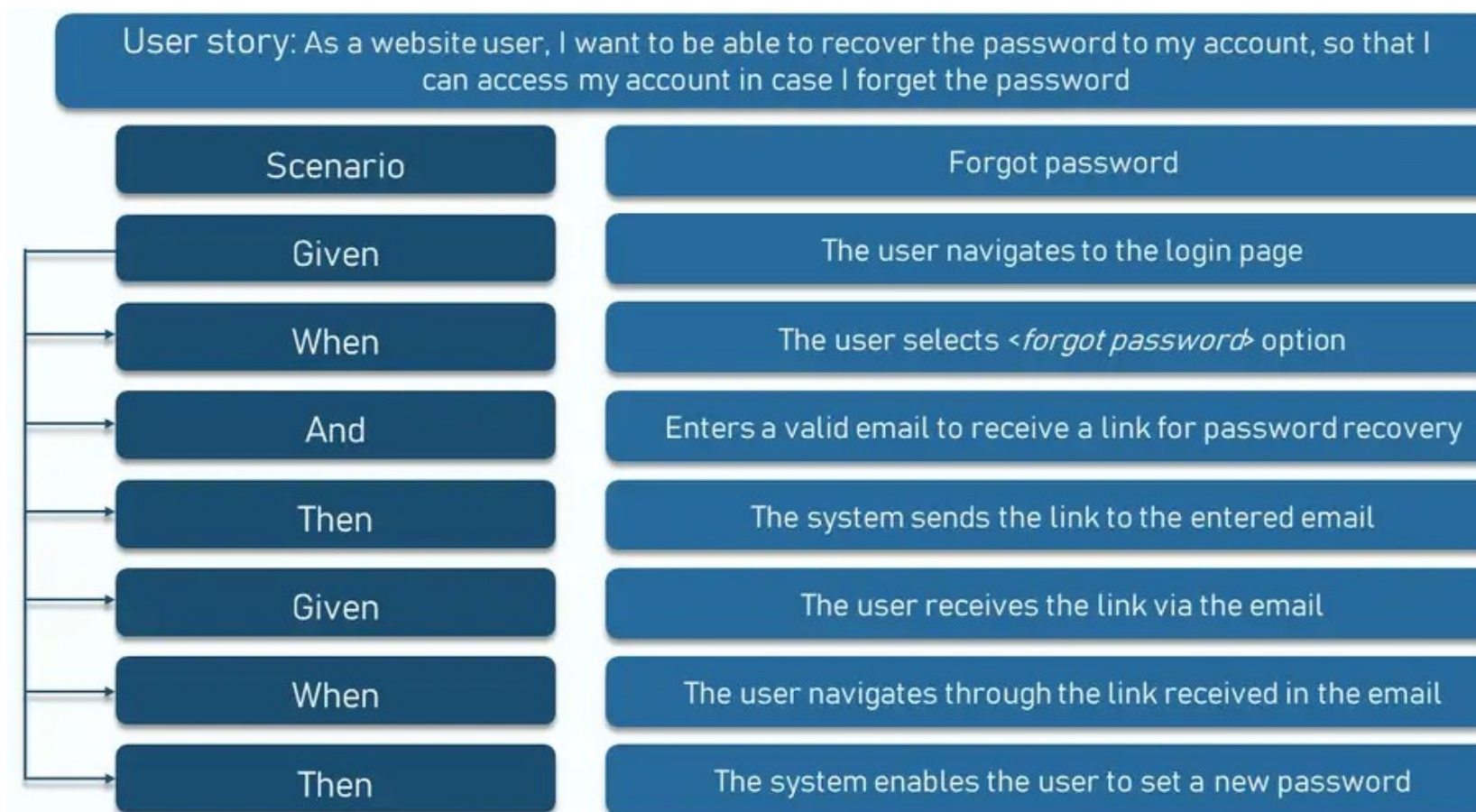
W — **WILL NOT HAVE**
Initiatives that are not priority for this specific time frame.

HOW DO WE KNOW IF A USER STORY IS COMPLETE?

- In agile, **acceptance criteria** refer to a set of predefined requirements that must be met to mark a user story complete

- A **product owner** may be responsible for writing acceptance criteria for the stories in the product backlog

SCENARIO-BASED ACCEPTANCE CRITERIA

User story: As a website user, I want to be able to recover the password to my account, so that I can access my account in case I forget the password

Scenario	Forgot password
Given	The user navigates to the login page
When	The user selects *<forgot password>* option
And	Enters a valid email to receive a link for password recovery
Then	The system sends the link to the entered email
Given	The user receives the link via the email
When	The user navigates through the link received in the email
Then	The system enables the user to set a new password

https://www.altexsoft.com/blog/acceptance-criteria-purposes-formats-and-best-practices/

MILESTONE 1 - REQUIREMENTS

Blackboard -> Project -> Proposal

Please read the documentation carefully for your first milestone!

A NOTE ON UML

- The Unified Modeling Language (UML) is a general-purpose, developmental modeling language in the field of software engineering that is intended to provide a standard way to **model the requirements and design of a system**.

- UML diagrams are heavily-associated with the Waterfall process (Big upfront design), but may have limited usage in processes like Scrum that focus on continuous changes.

READINGS

- Chapter 8-11. Software Engineering A Practitioner's Approach by Roger Pressman, 8th edition.

- Chapter 5. Software Engineering by Ian Sommerville. 10th edition

- 第8章 软件需求. 现代软件工程基础 by 彭鑫 et al.

NEXT

- Version Control System

TAO Yida@SUSTECH