# SE202
# Introduction to Software Engineering

**Lecture 2-2**
**Software Documentation**
**and Document Management System**

Pathathai Na Lumpoon

# Last Lecture –
## Software Development Process

- SDLC consists of:
  - Requirement Analysis
  - Design
  - Implementation
  - Testing
  - Deployment
  - Maintenance
  - Wrap up

# Topics - Software Development ….

- ***DOCUMENTs***

  need for software development

- ***PEOPLE***

  involve in a software project

# Software documentation


- Documentation is important at every step from the beginning to end of the development process.
- The main **goal** of effective documentation is to ensure that developers and stakeholders are headed in the same direction to accomplish the objectives of the project.

# Software Documentation

- Documents in software project?
  - Requirements documents
  - Use cases
  - Design documents
  - Test plans
  - User training material
  - And much more!
- Pages of documentation in a modest software project
  - Hundreds or even thousands pages
- Living documents that change over time
  - E.g. the requirements are allowed to change as the project progresses
  - Change control board is needed in the project.

# Document version

- Document version is a particular form of document that varies from other forms of the same document.
  - E.g.

# Tracking documents

- Tracking documents during development helps the team knowing what they are supposed to be doing.
  - The most recent version of the requirements tells the team what the application should do.
  - The most recent high-level and low-level designs tells the team if they are following the plan correctly.
  - The older versions of the documentation convey what changes were made, why they were made, and who made them.
- What if the team cannot find those documents or it takes a lot of time to check the documents? Solutions?

# Race condition

- Consider the following situation

SA Tony

Shared requirement document of an application producing quarterly reports of root beer demand

SA Peter

SA Tony and SA Peter open the document at the same time in order to change the requirement. SA Tony wants to change the font requirement to allow 12-point Arial and SA Peter wants to change some other part of the document.

Who do you think will be the winner to make a change?

How to prevent the race condition?

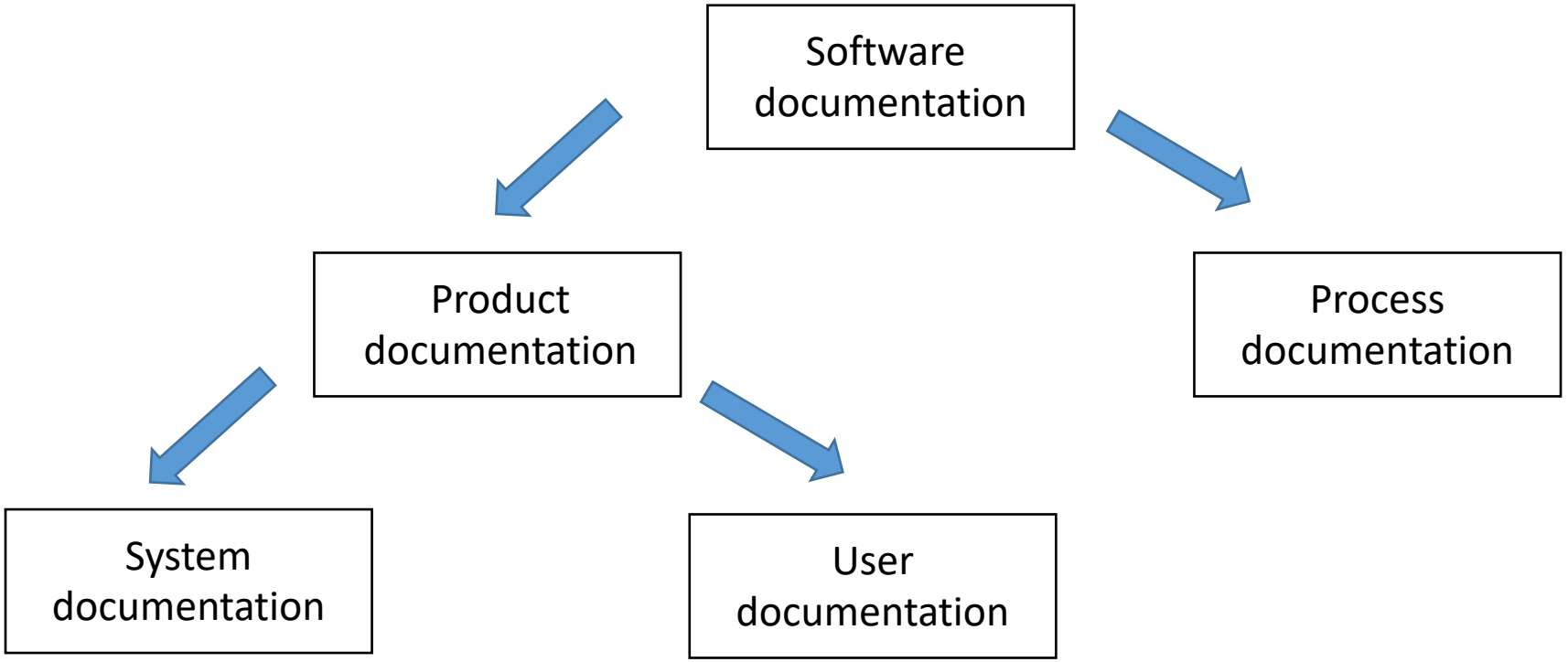
# Document management system (DMS)

- https://www.youtube.com/watch?v=bxHbP-q9InU
- People can share current and historical documents so that they can all view and edit them.
- Only one person can edit a document at a given time.
- You can fetch the most recent version of a document.
- You can fetch a specific version of a document by specifying either a date or version number.
- You can search documents for tags, keywords, and anything else in the documents.
- You can compare two versions of a document to see what changed, who changed it, and when the change occurred.
- Examples of DMS

# Program source code

- What is a different between **requirement and design documents** and **code** in the aspect of changing?
- Source code control systems is slightly different than other kinds of document control systems
  - If you add the word “comparison” to the beginning of a html paragraph, a source code control system like GitHub might detect a change of every line in the paragraph while a document control system like Microsoft word or Google Doc detects only the added word.
- GitHub helps software teams to collaborate and maintain the entire history of code changes.
  - https://github.com

# Documentation types



# Software documentation

- Process documentation
  - Represents all documents produced during development and maintenance that describe process.
  - E.g. standards, project documentation, such as project plans, test schedules, reports, meeting notes, or even business correspondence.
- Production documentation
  - describes the product that is being developed (System documentation) and provides instructions on how to perform various tasks with it (User documentation)

# Software documentation

- System documentation
  - Represents documents that describe the system itself and its parts.
  - E.g. requirements documents, design decisions, architecture descriptions, program source code, test plan and help guides.
- User documentation
  - Covers manuals that are mainly prepared for end-users of the product and system administrators.
  - E.g. tutorials, user guides, troubleshooting manuals, installation, and reference manuals.

# WHY do we need *Documentation* ?

**General requirements** of all software documentation

1. Should provide *for communication* among team members
2. Should provide enough information to management to allow them *to perform all program management* related activities
3. Should *describe to users how to operate* and administer the system
4. Should act as an information repository *to be used by maintenance engineers*

# General practices for all types of documents

- Write just enough documentation
  - Too much and poor documentation causes many errors and reduces efficiency in every phase of a software
- Documentation is an ongoing process
  - Keep documents up-to-date.
  - Use automatic version control
- Documentation is the collaborative effort of all team members
  - Documents should be shared in the team to get the feedback from all team member
- Hire a tech writer
  - Hiring an employee who will take care of your documentation.

# Stakeholders of Software Project

# Stakeholder?

- **stakeholder** A person, group, or organization that is actively involved in a project, is affected by its outcome, or can influence its outcome *by Karl Wieger's Software Requirements*

- *Who can be a stakeholder in the CMU course enrollment system?*

# Stakeholder categories

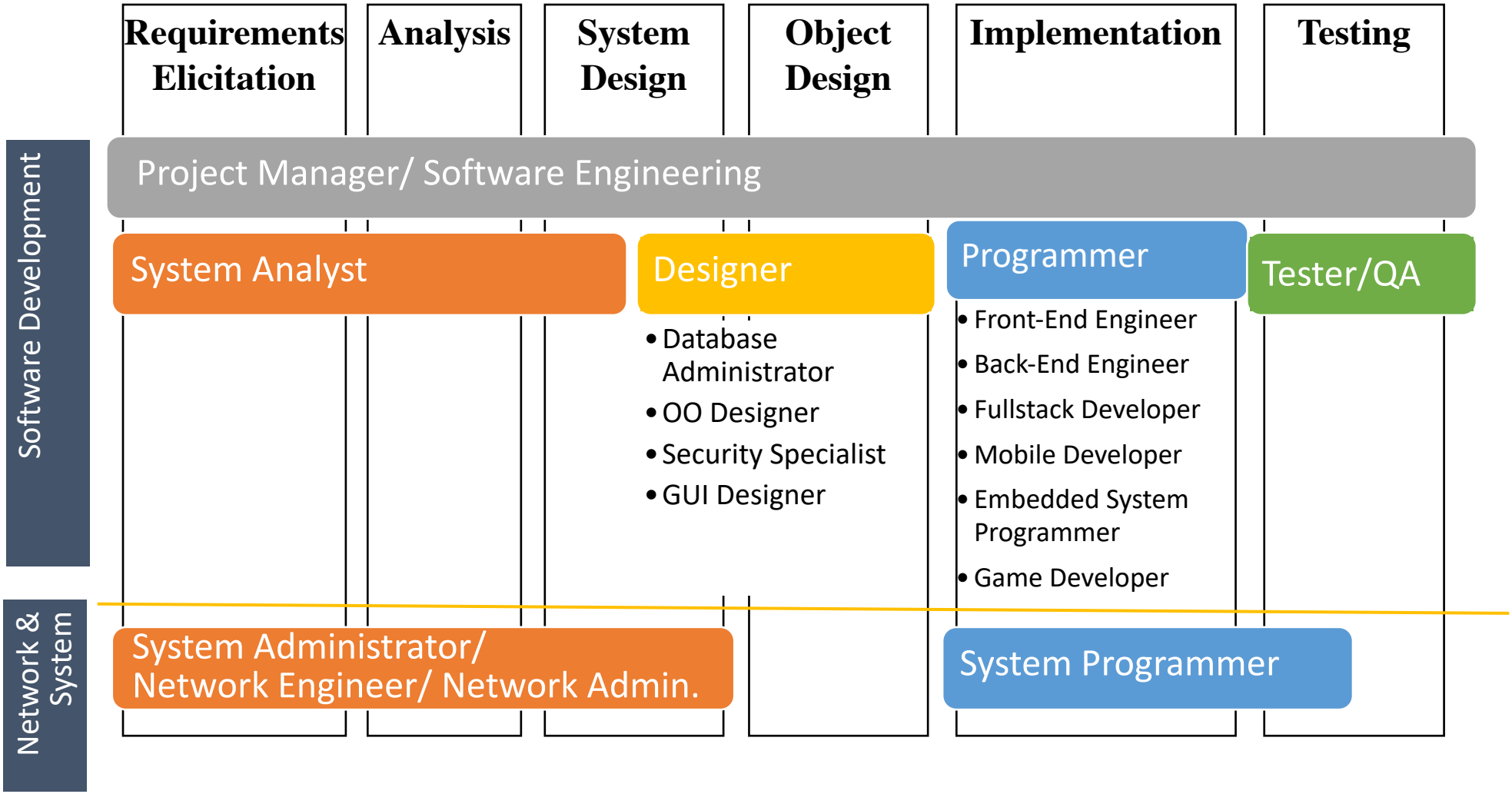
- Those who are involved in the project and work on it.
  - Project team (E.g. developers, architects, business analyst)
  - Management team (E.g. project manager)
  - Third-party companies
  - Support team (E.g. QA engineer, document)
- Those who are affected by the project and who will use its artifacts, including the results.
  - Customers
  - Heads and employees of functional units
  - End users
- Those who are not involved in the project, but because of their position or activities can influence it.
  - Top-managers of the company.
  - Owners of the company
  - Shareholders and creditors
  - Regulatory structures

# To success the project

- “It states that understanding and defining groups of people that can influence a business or a specific project makes it possible to clearly structure and optimize the management process.” by Strategic Management: A Stakeholder Approach

- Communication skill is the main tool used to influence stakeholders.

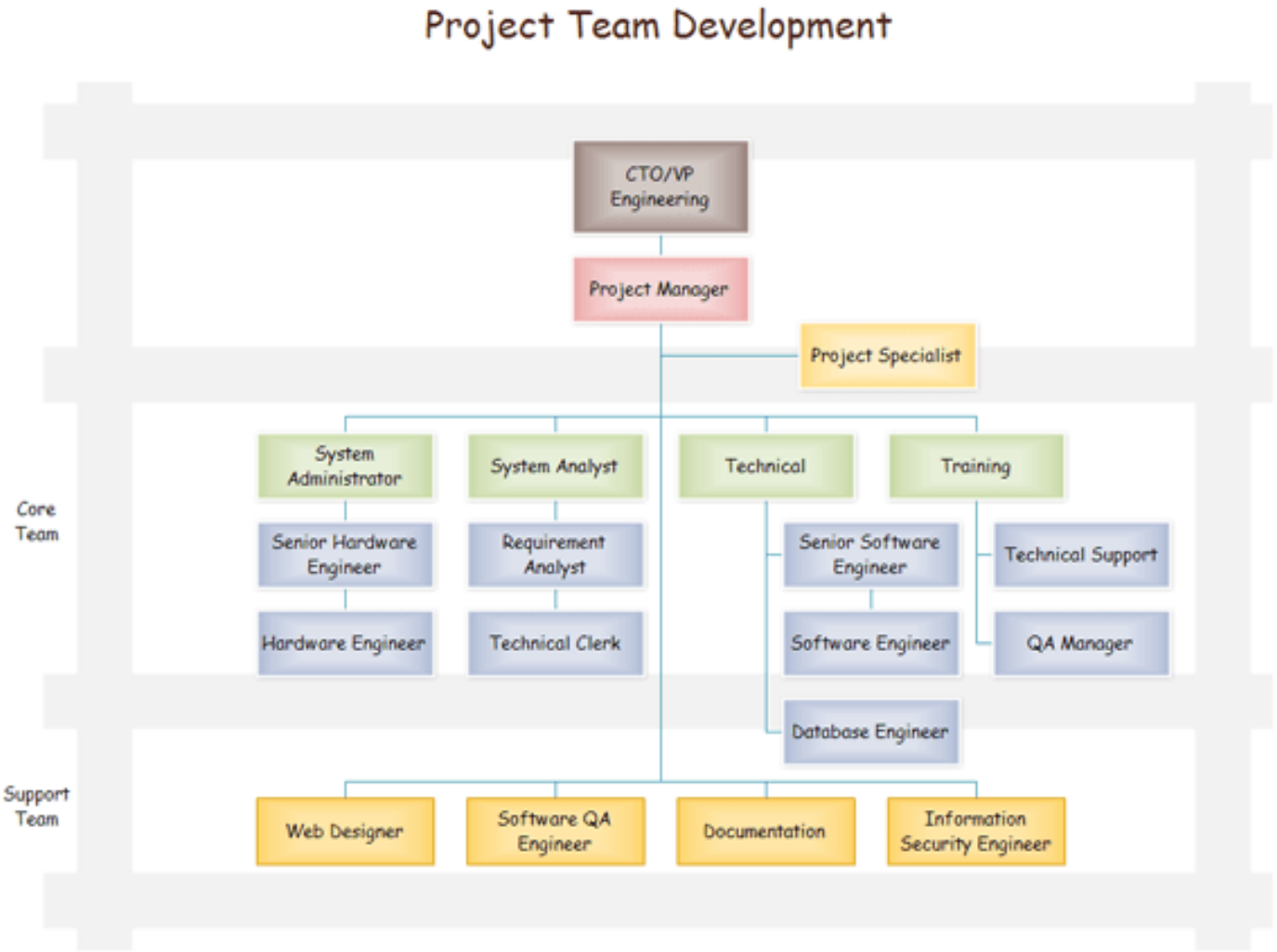
# Involvers of SW cycle phases



# Software Development Team

- Each position might not be fixed to one SDLC phases.
  - Scope of what position has to do depends on *Job Description*
- However, *most* of IT positions *require*
  - Logic and Skills of Programming
  - Data Structure of the implementing project
  - Problem solving skills
    - Problem analysis
    - Find out solutions by yourself
  - Social skills
    - Communicate
    - Team work

# Example of Project Team Development




https://i0.wp.com/edrawsoft.com/images/orgchart/project-matrix-orgchart.png