ĐẠI HỌC BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

Nhóm chuyên môn Nhập môn Công nghệ phần mềm

NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

Problems in Software Engineering

CONTENTS

1. Identify target audience of software
2. Balancing function, cost, and time
3. Risks of software project
4. Other problems

GOALS

By completing this session, learners are able to:

1. Issues in identifying the **target audience** of the software
2. The issue of **balancing function, cost, and time** in a software project
3. Potential **risks** during the project development process
4. Other issues

CONTENTS

1. Identify target audience of software

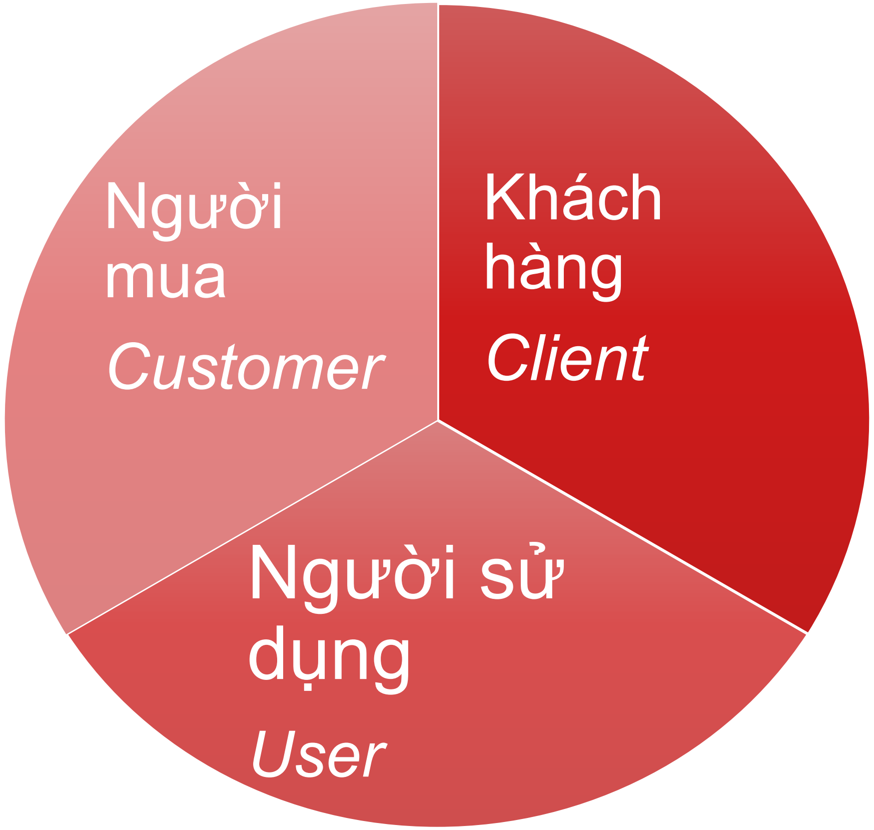
2. Balancing function, cost, and time

3. Risks of software project

4. Other problems

1. IDENTIFY TARGET AUDIENCE OF SOFTWARE

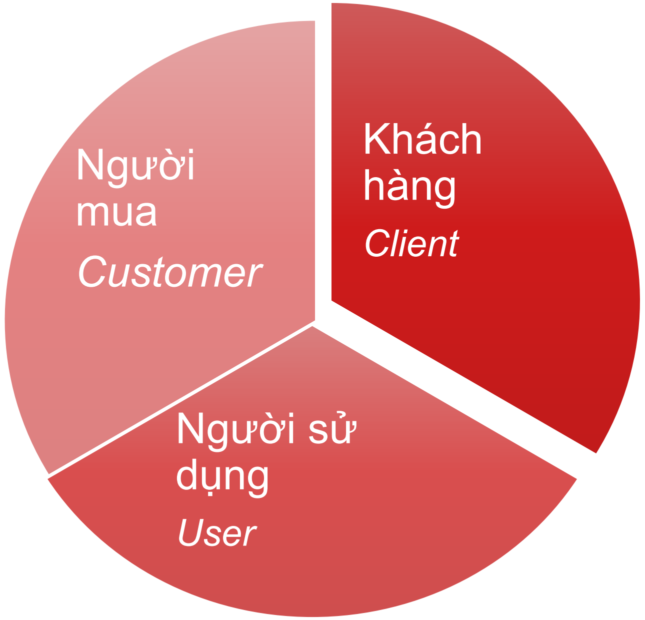
- Who are the intended target audiences of the software?
  - Khách hàng (**Client**)
  - Người mua (**Customer**)
  - Người sử dụng (**User**)



1. IDENTIFY TARGET AUDIENCE OF SOFTWARE

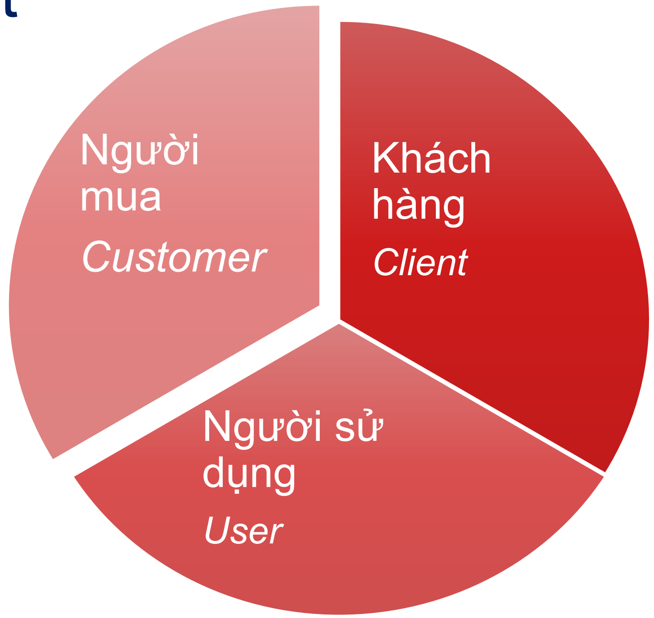
Client

- A person (or group of people) for whom the software development team creates the software
- The client provides resources (e.g., money) and expects to receive value in return through the product
- The client's success in their work may depend on the success of the software project



1. IDENTIFY TARGET AUDIENCE OF SOFTWARE

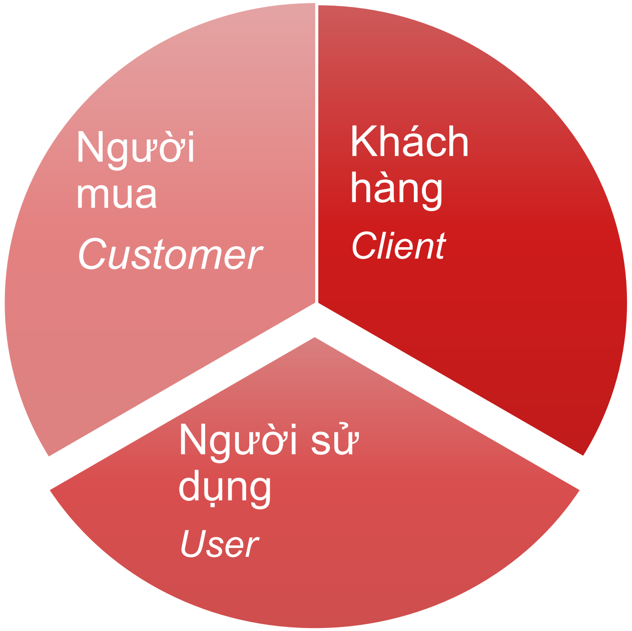
- **Customer**
  - The person who purchases the software or selects it for use by an organization or business
  - *Is a client the same as a customer?*



1. IDENTIFY TARGET AUDIENCE OF SOFTWARE

User

- The user, or end-user, is the person who actually uses the software.
- For personal software, the user and the customer may be the same person.
- In organization, the buyer and the user are often different.



Customer satisfaction is the most important measure of success in a software project

CONTENTS

1. Identify target audience of software

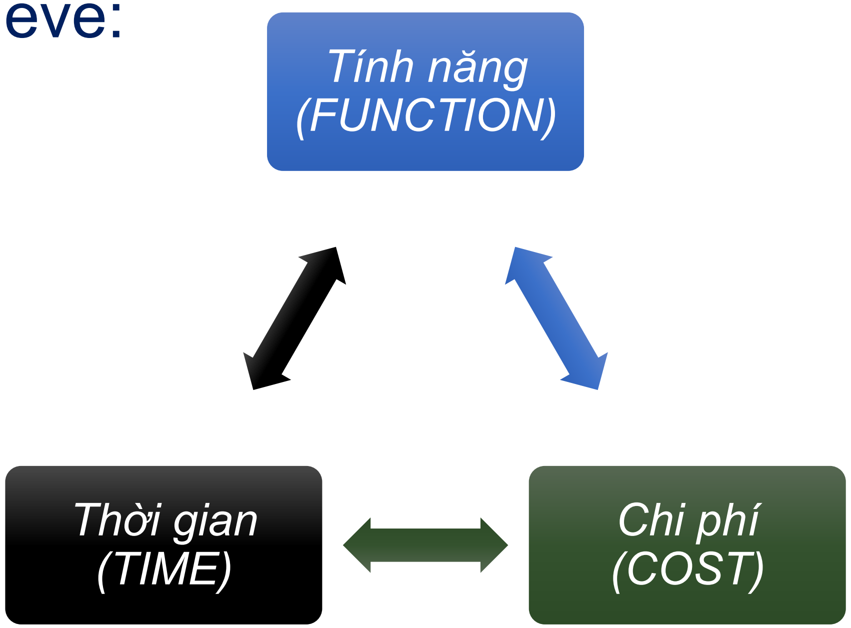
2. Balancing function, cost, and time

3. Risks of software project

4. Other problems

2. BALANCING FUNCTION, COST, AND TIME

- The development of a software project aims to achieve:
  - A product that functions as expected (**function**)
  - Within budget (**cost**)
  - Delivered on time or ahead of schedule (**time**)



- Reality:
  - Every additional feature increases the cost of development, testing, maintenance, etc

□ Balance between function, cost, and time

CONTENTS

1. Identify target audience of software
2. Balancing function, cost, and time
3. **Risks of software project**
4. Other problems

3. RISKS OF SOFTWARE PROJECT

- Risk
  - A large portion of software functionality is **wasted** (~50% is never used)
  - Many software projects fail because developers build the **wrong** software
  - ...
- The software development team must:
  - Understand what the customer expects from the software
  - Understand what the customer’s organization expects from the customer
  - Understand what both the customer and the user expect from the software

3. RISKS OF SOFTWARE PROJECT

- Risk mitigation
  - Feasibility study (Should the project be initiated?)
  - Separate requirements (what the customer wants) from design (how developers fulfill those requirements)
  - Milestones (how developers report or demonstrate progress to the customer) and releases
  - User testing and acceptance testing (how the customer verifies whether the software meets the requirements)
  - Delivery (ensuring the customer receives a product package that can be operated and supported over the long term)

3. RISKS OF SOFTWARE PROJECT

- Risk mitigation
  - Transparency: Responsible individuals must know what is happening
  - Issues (identified by management) rely on others to report progress or difficulties
- Problem: Software developers often
  - Are not good at assessing progress
  - Tend to be overly optimistic about the project's advancement
  - Consider reporting to be a waste of time
  - …

CONTENTS

1. Identify target audience of software
2. Balancing function, cost, and time
3. Risks of software project
4. **Other problems**

4. OTHER PROBLEMS

- Failure to meet business needs
- Failure to satisfy requirements
- Lack of integration between modules
- Difficulty in maintenance
- Late detection of defects
- Poor user experience quality
- Low software development productivity
- Lack of team coordination efforts
- Issues in building and releasing the product

4. OTHER PROBLEMS

- No clear method to describe customer requirements

  □ Leads to issues after delivery

- For large-scale software, rigid specification documents

  □ Make it difficult to accommodate changing user needs

- Inconsistent design methodologies

  □ Designs are done in individual ways □ Reduces software quality

- No standard for documenting the software production process

  □ Unclear specifications □ Decreased software quality

4. OTHER PROBLEMS

- No verification of software **correctness** at each stage, only testing at the final stage

□ Often results in late delivery

- Undervaluing the design process

□ Reduces software quality

- Neglecting software reuse

- Most activities in the software development process are performed manually

□ Reduces labor productivity

4. OTHER PROBLEMS

- Inability to prove the **correctness** of the software
  - □ Reduces software reliability
- Standards for good software cannot be measured **quantitatively**
  - □ Makes it impossible to evaluate whether a system is correct
- Large workforce investment in maintenance
  - □Decreases employee productivity

4. OTHER PROBLEMS

- ▪ Prolonged maintenance work

□ Reduces documentation quality and negatively impacts other tasks

- ▪Loose project management

□ Leads to unclear software production scheduling

- ▪No standards for estimating manpower and budgeting

□ Results in project delays and cost overruns

SUMMARY AND OUTLOOK

1. The lesson has introduced learners to several key **problems** that need attention in software engineering, such as identifying the **target audience** of the software; **balancing function, cost, and development** time; **risks** in software development; and other related concerns.
2. Following this lesson, learners will explore concepts of the software **lifecycle** and software **development processes**.

NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

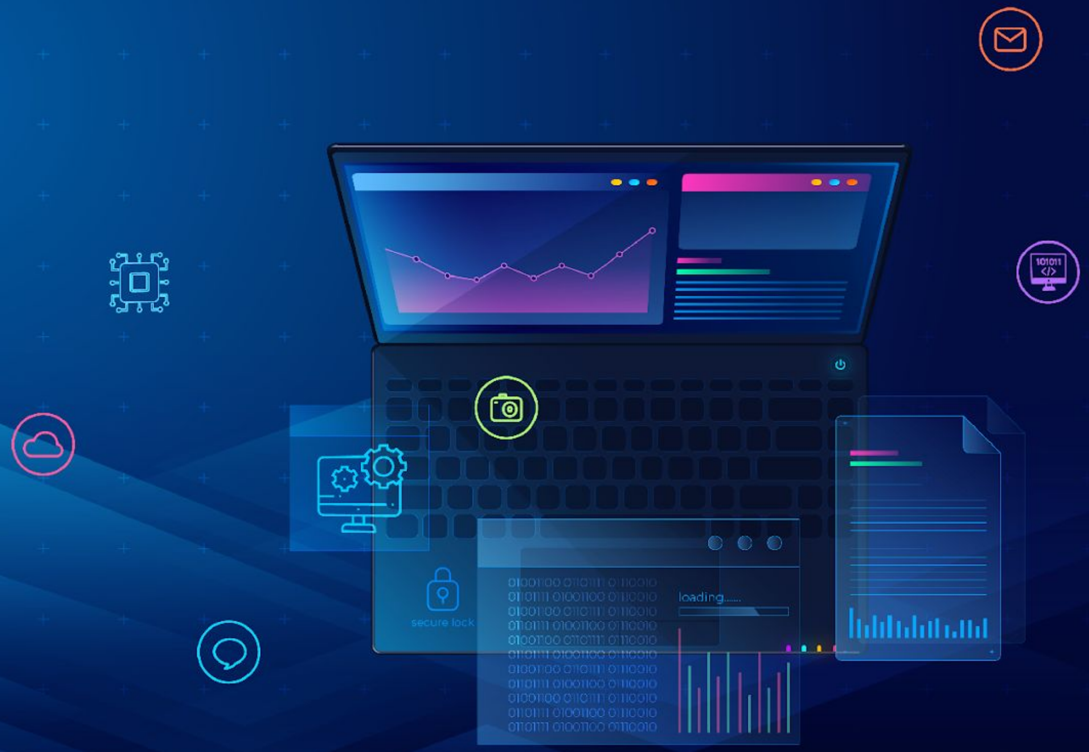
Các vấn đề trong Công nghệ phần mềm

Biên soạn:

TS. Trịnh Thành Trung

Trình bày:

TS. Trịnh Thành Trung



NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

Bài học tiếp theo:

Tổng quan vòng đời phần mềm

Tài liệu tham khảo:

[1] R. Pressman, Software Engineering: A Practitioner’s Approach. 8th Ed., McGraw-Hill, 2016.

[2] I. Sommerville, Software Engineering. 10th Ed., AddisonWesley, 2017.

[3] Pankaj Jalote, An Integrated Approach to Software Engineering, 3rd Ed., Springer.

[4] Shari Lawrence Pleeger, Joanne M.Atlee, Software Engineering theory and practice. 4th Ed., Pearson, 2009

KẾ HOẠCH GIẢNG DẠY

Tuần	Chương	Video
1	Vai trò của công nghệ phần mềm	C1.3-SERoles
1	Các vấn đề trong công nghệ phần mềm	C1.4-SEProblems
2	Tổng quan vòng đời phần mềm	C2.1-Intro