1. Course Code

2207

2. Course Title

F50e: Introduction to Computer Software

3. Teacher

ITO, Mamoru

4. Term

Fall 1

5. Course Requirements (Courses / Knowledge for this course) and Important Information

None

6. Course Overview and Objectives

You will learn the fundamentals of the processes and techniques necessary for developing and deploying software systems. The course also covers the latest technologies that are driving the digital transformation of our society. By the end, you will have a solid grasp of software characteristics and their social context. You will also refine your decision-making and behavioral skills to solve problems that may arise in software projects. Students will learn to respond flexibly to trends in software technology development.

7. Course Outline

1. Basic computer principles
2. Data types and data structures
3. Digital Transformation - Cloud computing and web design
4. Digital Transformation - Machine learning and blockchain
5. Software engineering and ethics
6. Software development processes
7. Software requirements
8. Software design
9. Software testing and quality
10. Object-Oriented concepts
11. Introduction to UML - Behavior diagrams
12. Introduction to UML - Structure diagrams
13. UML modeling exercise
14. UML modeling exercise
15. Modeling exercise presentation
16. Term-end examination (multiple-choice and open-book format)

8. Textbooks (Required Books for this course)

None

9. Reference Books (optional books for further study)

H. Washizaki, eds., *Guide to the Software Engineering Body of Knowledge (SWEBOK Guide), Version 4.0* , IEEE Computer Society, 2024; www.swebok.org.

10. Course Goals (Attainment Targets)

(1) Apply software development life cycles and process models to software projects.
(2) Utilize basic software analysis and design techniques.
(3) Develop the practical decision-making skills necessary for software projects.
(4) Understand the social environments surrounding software development.
(5) Analyze the ethical issues involved in software development.
(6) Be able to respond flexibly to trends in software technology development.
(7)
(8)

11. Correspondence relationship between Educational goals and Course goals

Educational goals of the school			Course Goals
High level ICT skills	Basic academic skills		(1), (2)
	Specialized knowledge and literacy		(1), (2)
Human skill (Tankyu skill)	Ability to continually improve own strengths		(2)
	Ability to discover and resolve the problem in society	Problem setting	(3), (4)
		Hypothesis planning	(3), (4)
		Hypothesis testing	
		Practice	
	Fundamental Competencies for Working Persons	Ability to step forward	(3), (6)
		Ability to think through	(3), (4), (6)
		Ability to work in a team	(1), (6)
Professional ethics			(3), (4), (5), (6)

12. Evaluation

Goals	Evaluation method & point allocation					
	Examination	Quiz	Reports	Presentation	Deliverables	Reflection
(1)	○	○		○	○	○
(2)	○	○		○	○	○
(3)	○	○	○	○		○
(4)	○		○	○		○
(5)	○		○	○		○
(6)	○	○	○	○	○	○
(7)						
(8)						
Allocation	30	25	10	10	20	5

13. Evaluation Criteria

Examination	A multiple-choice exam evaluates students' comprehension, application, and thinking skills regarding the course content. Since it is an open-book exam, students can use their own materials and are not required to memorize information.
Quiz	Multiple-choice quizzes evaluate students' comprehension, application, and thinking skills regarding the course content.
Reports	Student reports are evaluated based on their accuracy and completeness, clarity and conciseness, and adherence to instructions.
Presentation	Student presentation is evaluated based on its accuracy and completeness, clarity and conciseness, and conformity to objectives.
Deliverables	Student deliverables are evaluated based on accuracy, completeness, clarity, conciseness, and adherence to instructions.
Reflection	Students' reflection journals are evaluated based on their depth and quality.

14. Active Learning

Hourly percentage of active learning within the whole class time		40%
1	Active learning such as problem solving assignment using the knowledge and skills acquired in class.	All the time
2	Active learning such as group works and discussions.	All the time
3	Outcome presentations and feedbacks.	Sometimes
4	Students actively make decisions on how the class should be conducted.	Not at all

15. Notes

- Course materials will be provided on Moodle or Google Classroom.
- Although you may attend classes online, we strongly encourage you to attend in person whenever possible.
- If you attend classes online, you may be asked to turn on your video.
- Some class sessions will be recorded and made available on demand for the duration of the course.

16. Course plan

(Notice) This plan is tentative and might be changed at the time of delivery

Lesson 1: Basic computer principles — Lecture/Discussion 90 min

Software runs on a computer. A basic knowledge of computers will improve your understanding of software development. In this lesson, we will learn how a computer works.
- Introduction
- Computer organization (CPU, memory, clock)
- Von Neumann architecture
- Memory hierarchy
- Program performance equation

Lesson 2: Data types and data structures — Lecture/Discussion 90 min

A data type is a format that defines how data is handled, and a data structure is a representation format for efficiently storing and organizing data. In this course, students learn the basics of data types and data structures.
- Notation
- Outline of data types
- Address space
- Typical data structures (arrays, lists, stacks and queues, tree structures)

Lesson 3: Digital Transformation - Cloud computing and web design — Lecture/Discussion 90 min

A variety of digital technologies are being used to drive digital transformation (DX). This lesson provides an overview of DX, followed by the basics and latest trends in cloud computing and web technologies.
- What is Digital Transformation (DX)?
- Strategy first
- Cloud computing
- Web technologies

Lesson 4: Digital Transformation - Machine learning and blockchain	Lecture/Discussion 90 min

This lesson focuses on machine learning and data science, among the various digital technologies used to drive digital transformation (DX). We will discuss the growing opportunities and challenges of these technologies.
- Basics of Machine learning and data science
- AI and society
- Blockchain technologies
- web3

Lesson 5: Software engineering and ethics	Lecture/Discussion 90 min

With the popularization of computers, software is becoming increasingly important. We will discuss the reality surrounding software development after understanding the characteristics of software and learning the necessity of software engineering.
- Features of software
- Importance of software
- Environment surrounding software development
- Role of software engineering and ethics

Lesson 6: Software development processes	Lecture/Discussion 90 min

A "process" can be defined as "a set of related or interacting activities that transform inputs into outputs. A good process is needed to produce good outputs. We will get an overview of software life cycle process models and the importance of process improvement.
- Definition of software process
- Life cycle models
- Present situation and issues on software process
- Meaning of software improvement

Lesson 7: Software requirements	Lecture/Discussion 90 min

The role of software engineers is to realize the requirements of customers and users by using the software. But their requirements are sometimes ambiguous and lack consistency. We should acquire their requirements exhaustively and analyze them systematically. We will marshal the concepts of requirements and flow of requirement analysis.

- Difference between needs wants and demands
- Functional requirements and non-functional requirements
- Requirements analysis techniques
- Requirements modeling

Lesson 8: Software design	Lecture/Discussion 90 min

The role of software engineers is to realize the requirements of customers and users by using the software. But their requirements are sometimes ambiguous and inconsistent. We should collect their requirements thoroughly and analyze them systematically. We will review the concepts of requirements and the flow of requirements analysis.
- Design viewpoints
- Data Flow Diagram (DFD)
- State Transition Diagram (STD)
- Entity Relationship Diagram (ERD)

Lesson 9: Software testing and quality	Lecture/Discussion 90 min

Software testing and quality management are becoming increasingly important because software defects have a significant impact on society. In this lesson, we will learn the basic concept of software testing and quality management.
- What is software testing?
- Major testing techniques
- Quality and grade
- Software quality model
- Quantitative quality management

Lesson 10: Object-oriented approach	Lecture/Discussion 90 min

The object-oriented methodology is gaining popularity in connection with the increasing size and complexity of software. This methodology is used not only for programming, but also for requirements analysis and software design. This lesson focuses on object-oriented analysis and design using UML modeling.
- A brief history of object-oriented methodology
- Object-oriented model
- Object-oriented principles
- Object-oriented analysis

Lesson 11-12: Introduction to UML	Lecture/Exercise 180 min

UML is becoming widely used with object-oriented technology. UML stands for Unified Modeling Language, a useful tool for analyzing and designing complex software systems and businesses. We will learn how to describe important diagrams.
- What is modeling?
- What is UML?
- Behavior diagrams
- Structure diagrams

Lesson 13-14: UML modeling exercise	Exercise: 180 min

UML is used to model businesses and processes and to analyze, design, and implement software systems. In this lesson, we will focus on different businesses, and through group work, we will model each business using UML.

Lesson 15: Modeling exercise presentation	Presentation 90 min

Each group will present the results of their UML modeling. All group members should prepare presentation materials to ensure that everyone has the opportunity to present.

Term-end Examination	Examination: 90 min

A timed, multiple-choice exam is administered to assess students' understanding, application, and critical thinking of course content. This is an open-book exam, allowing students to use their own study materials.