BITS College

Curriculum of Undergraduate Program in *Software Engineering*

Revised July 2024

Addis Ababa, Ethiopia

Promoting excellence in learning and teaching

PROGRAM SUMMARY

Name of the Degree Program:	B.Sc. in Software Engineering
Awarding Institution	BITS College
Standard Period of Study:	5 Academic Years with 10 Semesters
Total Credit Hours (Cr.):	179 Cr hrs. (304 ECTS)
Commencement of the Revised Program:	2016 E.C. (2023/24 G.C)
Remark:	If there is any inconvenience in the curriculum regarding course name, course code, course weight and other related issues, the semester course breakdown will govern.

Table of Contents

1. Introduction

1.1 BITS College

BITS, an abbreviation formed from the first letters of Business, Innovation, Technology, and System, is a private higher learning institution with a vision of promoting excellence in the production, growth and dissemination of advanced scientific knowledge through teaching and research. The College is conceived, established and run by caring and committed educators and innovators who seek to improve the quality of higher education in the country through the introduction of innovative and enlightened education programmes that help students realize their potential. It aims at realizing this by engaging a management team experienced in education and business, a dedicated team of faculty and staff, well-designed academic programmes, world class educational facilities and cutting-edge technologies. The senior management team comes with over 30 years' combined experience in teaching at tertiary level at Addis Ababa University (AAU), holding senior management positions at AAU (education management), unique and proven track record in corporate management in technology (IT service) industries.

Amongst the founders of the College is a focused and well-reputed system development and training company, with proven track record in business process management and enterprise software development and support. Founded in 2012, the IT Company mainly involves in the design and development of innovative and high-quality web-based business applications for the logistics, construction, and health sectors. In fact, BITS had its genesis in this IT Company.

BITS plans to engage in mutually rewarding collaborations and strategic partnerships with national, international, public, and private higher learning and research institutions so as to grow and become a full-fledged university that offers undergraduate and graduate degree programs in business and technology related fields.

Currently, the College offers three academic programs, namely:

- Bachelor of Science Degree in Software Engineering
- Bachelor of Science Degree in Information Technology and Systems
- Master of Science Degree in Information Technology Management

The undergraduate program in software engineering started in June 2021 per the accreditation given from the Education and Training Authority (ETA) in October 2020. Since then it has accepted three batches of students. Per the review of the needs assessment and stakeholder engagement analysis during the course of the three years, the college has made major revisions in its curriculum which was also assessed by the ETA assessors. Moreover, per the requirement of the ETA, the software engineering undergraduate program curriculum is also organized in five years effective 2024/25 academic year (2017 E.C)

The main purpose of this document is to present the required narratives for the five years **undergraduate program in Software Engineering**. Accordingly, the document is organized as follows. The remaining part of this section presents the rationale for the undergraduate program in Software Engineering, the stakeholder engagement reviews and needs assessment conducted over the past three years. The second section of the document presents the curriculum. The third section details the course description. Resource requirements of the program are presented in section four. Section five presents the course offering schedule for both the regular and extension programs.

1.2 Rationale

ICT has taken the central stage in almost every aspect of human endeavor. It helps improve the efficiency and effectiveness of services offered to customers, and enhance business processes, managerial decision making, and workgroup collaborations, which strengthens competitive positions in rapidly changing and emerging economies. Particularly software applications have profoundly transformed markets, industries and the society in general. Not only is the dependence on software increasing but the character of software production itself is changing and with it the demands of the industry. Furthermore, with the huge investment in business industries such as Banking and Telecom, there is a greater demand for software engineering professionals of world standard. Specializations in various technical knowledge such as requirements engineering, architecture design, programming/coding and are in demand. As such, both the software and business industries expect students to be educated in courses and projects that are professionally relevant and that prepare them well for the work place.

Needless to say, that the Ethiopia's future lies in educating the citizens to the highest possible standards. In order for the country to reach its economic and social goals, a thriving and

successful higher education system is essential. The increasing enrolment and graduates in recent years also indicate the commitment in this country to further expand and modernize tertiary level education - to provide greater opportunities for all citizens. We also observe in the job market, that a college degree is becoming the preferred currency of the job application processes more and more - those without degrees are being given less and less preference by employers.

Currently, almost all of these higher learning institutions have one or more IT related undergraduate programs. Despite such encouraging developments of increasing the number of Colleges, programs and college degree holders, serious concerns are being expressed with regard to the quality of graduates.

- There is widespread dissatisfaction among both graduates and their employers on the performances of the graduates in the work area.
- The enrolled and graduates feel not necessarily better educated in employable skills, problem solving skills, critical thinking skills, etc.
- Employers feel that current graduates are deficient in thinking and problem-solving skills and hence inadequate for the demands of the workplace.
- In the case of IT graduates, for instance, graduates lack the ability to link technology and information systems with business processes and strategic objectives of organizations.
- There is a growing awareness among employers that graduates entering the workforce with such deficiencies would have a great repercussion on the ability to be competitive in a global marketplace.

Taking cognizance of this, as of recent, the need to introduce initiatives to improve/increase the quality of education is being advocated widely. Deliberations are underway at various forums on the whys and wherefores of the deficiencies. Among the issues under consideration are: revisiting college entrance preparations and exams; exploring ways and means of considering employable skills in the design and delivery of curricula; redesigning the national education roadmap, et cetera.

To this end, in the wake of the numerous challenges facing education in the country, and motivated by some of the national initiatives in this connection, BITS College is established to make its share of contribution to the ongoing efforts of quality improvement. We seize this

chance to address the challenge of providing education that meets high quality standards and whose contents are aligned to the needs of the country's economy and society. More specifically, the aim is to prepare students in the theory and methods of systematic and rigorous construction of software for industrial, scientific and commercial applications.

The revised undergraduate program in **software engineering** is a step in this direction.

1.3 Summary of Needs Assessment

In such dynamic area of studies as software engineering and information technology, a lot of changes and developments take place every time. As such, a regular periodic revision and updating of the curriculums are necessary to incorporate and reflect such changes and developments.

In this regard, the curriculum for the Bachelor of Science Degree in Software Engineering has been revised based on series of discussions that have been made with existing students as well as stakeholders. As further justification of the undergraduate program under consideration, a detailed needs/market assessment has been carried out previously, during the establishment of the program back in 2019. While the detailed needs assessment report is provided in a separate document, the following is a summary made from the discussion with enrolled students of the software engineering program as well as the stakeholder engagement analysis.

Stakeholders in general agree that the benefit of training in software engineering is exceedingly important. Software engineering methodology is gaining importance as the traditional software development is no more adequate. There is a growing need for user based and context based software engineering methodologies to generate usable software. Further, the trend is changing on requirement specification techniques. This has led to an increasing need for modern software engineering methodologies.

The majority of stakeholders claim that they need software engineering expertise in their organization's skill sets. They further noted that so far they rely on limited number of employees who also do not possess the required skills to develop usable software. This makes the demand for proper software engineering education to grow rapidly. Particularly, respondents from software development companies stated their need for full stack developers.

In addition, respondents noted that so far they were relying mainly on any graduates in computer science/IT areas. However, a specialized professional well versed with software development is important.

The results from the revised needs assessment also affirm that companies prefer to employee high level expertise in software development and IT related tasks to go into the work environment directly instead of wasting their time for trainings that should have been undertaken during their undergraduate years. They also noted that they are still in need of specialized experts in IT who are capable of handling the companies IT and software requirements.

Both students and stakeholders are well aware of the software development and the networking and system administration capabilities required in the future. Reflecting on the limitations of the existing IT professionals, they suggest the following skill needs: Agile Development; Preparing and organizing data repository; handling large data set; data center building; system and network administration; mobile application; data analytics and visualization and problem solving skills.

Furthermore, stakeholder and students recognized the following limitations:

- Full stack developer knowledge is limited
- Big data analytics capacity is limited
- There is a need for training in data science
- Limited skills of juniors to work with high level Fintech professionals in the banking industry.
- Limited skills of software development professionals to work and assist in teams involved high level organizations such as the banking industry.

With regard to modalities of training, respondents welcomed both long term degree programs including B.Sc. and M.Sc. In addition, Respondents recommended the undergraduate program to be offered both in regular and extension.

The revised undergraduate program in **Software engineering** takes into consideration the concerns raised. It is also revised based on the current state of knowledge as reviewed from existing literature and experiences shared with high level professionals in the field.

The remainder of this document, therefore, presents the revised curriculum.

2. Bachelor of Science in Software Engineering

2.1 Program Objective

The Bachelor of Science in Software Engineering (BSc. SE) intends to produce quality, world-class graduates in this rapidly developing discipline. The degree program has been specially designed in response to industry demand to produce graduate software engineers with software systems project capability. It is observed that graduates who can offer skills in these areas are in demand by business and government organizations concerned with software development. Graduates will be prepared for careers across all industries as Software Engineers, Applications Developers, Programming Specialists, and Systems Analysts.

2.2 Graduate Profile

Software systems are the cornerstones of all modern business. Such systems are often complex and long lived, and must be robust and adaptable. By studying software design and production techniques, this degree program will equip students with the skills needed to follow a career specifying and developing these systems, and other computer-based solutions. Students gain not only knowledge and practical experience of the latest technologies, but also a grounding in the underlying principles of the subject.

After obtaining the degree in software engineering, graduates will have the following profiles

(i) Knowledge and understanding of:

- theories, practices and principles of software engineering
- theories, principles, processes and recommended techniques of requirements gathering
- theories of software project management
- principles of software project management
- best practices of software project management
- issues affecting the industry and its technologies.
- computers and communication systems, including basic, network design, database development, implementation and management;
- programming languages and algorithm development to solve real world problems;
- importance of writing clear, understandable and maintainable code
- issues related to software quality and assurance

- researching, designing context-based software programs
- creating, maintaining, auditing and improving systems to meet particular needs,

(ii) Practical Skills:

- design and write code for operating systems and software to ensure efficiency.
- integrate existing software products for incompatible platforms to work together
- maintain systems by monitoring and correcting software defects
- ability to communicate with clients, colleagues and management to explain complex issues clearly and concisely
- work well in groups and understand the various roles played by fellow team members
- strong writing and communication skills,

(iii) Attitudes and Values:

- make contributions to the further development of the discipline
- have a positive and responsive attitude towards the value of their profession in general and software systems and resources in particular.
- have good personal confidence in their jobs and professional activities;
- have the sense of co-operation, honesty, loyalty, etc.;

2.3 Admission Requirements

Ethiopian Students

Applicants should meet the following requirements. For admission to BITS College.

(i) A graduate of an accredited high school with the required pass mark set by the national examination agency to enter higher education AND **the necessary pass mark in the College Entrance Examination.**

OR

(ii) A TVET Graduate approved for entry to higher education with official COC AND **the necessary pass mark in the College Entrance Examination**

OR

(iii) A graduate of an accredited higher education and **the necessary pass mark in the College Entrance Examination**

Foreign Students

- Admission of foreign students is based on the equivalence established by the Ministry/Higher Education Relevance and Quality Assurance Agency.

2.4 Structure of the Program

The program has 39 core courses, 3 electives and 12 supportive courses that are designed to be completed within 5 years of study. There are a total of 10 semesters (i.e. two per academic year) and each semester has 16 weeks.

2.5 Duration of Study

The duration of study for undergraduate degree regular program in Software Engineering is five years. In the case of evening programs, the duration of study shall be six to seven years.

A student who withdraws for valid reasons shall be granted readmission within six years after the date of withdrawal and failure to apply for readmission within this period of time shall entail dismissal for good.

2.6 Assessment and Evaluation

Examination is the main component of the evaluation methods. Final examination (50%), continuous assessment (50%) is favored for almost all the courses together with other methods stated in each course.

The traditional assessment methods, paper exam, will be used in most of the courses. Based on the nature of the course, the following assessment methods might also be used.

- Peer Assessment
- Progress Assessment by advisors for Projects
- Presentation for course-based projects and senior project
- Laboratory practical test

Whenever the practical part of a course is crucial for declaring competency of a course, the students should score a satisfactory result in the practical assessment of the course. The student shall fail the course if he/she fails to meet the minimum threshold of the practical assessment.

Details are also given under individual courses.

2.7 Course Exemption

A student may be exempted from a core course based on previous study, provided that he/she passes the exam set by the College for the specific course. An exemption from a course has no credit value towards a degree. Exemptions from support courses may be granted in cases where students have already covered the work at any accredited higher learning institution with an appropriate level of performance (a minimum grade of C).

Students would be granted exemption for up to 20% of courses they have taken in their respective program of study.

2.8 Grading System

Examinations are graded on the following letter grading system, with corresponding points.

Raw Mark[1]	Letter Grade	Grade Points	Description	Class Description
[95, 100)	A+	4.00	Excellent, Exceptional	First Class with Great Distinction
[85, 95)	A	4.00	Excellent - Outstanding Performance	
[80, 85)	A-	3.75	Excellent	
[75., 80)	B+	3.50	Very Good – Better than Average Achievement	First Class with Distinction
[65, 75)	B	3.00	Very good	
[60, 65)	B-	2.75	Very Good – less than average achievement	First Class
[55, 60)	C+	2.50	Good Achievement	Second Class
[50, 55)	C	2.00	Average Achievement	Second Class
[45, 50)	C-	1.75	Lower than average achievement	Lower Class
[40, 45)	D	1.00	Fail	Lower Class
< 40	F	0.00	Fail	Lower Class
	AU	Neutral	Successfully audited a course – no grade is assigned	
	CO	Neutral	Course continued in the following semester and grade assigned at that time – not included in grade – point average calculation.	
	W	Neutral	Withdrawn: Student has withdrawn from the course – no academic penalties	

[1] *The square bracket - **[** - indicates that the number is included in the respective range; The open bracket -) - indicates the number is excluded in the respective range.*

Raw Mark[1]	Letter Grade	Grade Points	Description	Class Description
	DO	Neutral	Drop Out: A student has not withdrawn from a program in accordance with the withdrawal procedures set forth by the College or has not produced evidence justifying his failure to sit for the examination	
	NG	Neutral	No Grade for some reason – This grade will be changed to F unless an appropriate reason comes or grade given in 6 weeks time.	
	I	Neutral	The student has not yet completed all requirements to receive a grade. The instructor has to write the reason why the grade of I is given. This grade will be changed to F unless an appropriate reason comes or grade given in 6 weeks time.	

2.9 Assignment of Course Codes

The course code has Four alphabets and three-digit numbers like SWEN101. The four alphabets code indicates the name of the program with all capital letters. For instance, SWEN indicates abbreviation of the program of Software Engineering.

The course codes are made in the following format:

- SWENXXX where:
  - "SWEN" represents the short form of the program name for courses in the software engineering.

Following the same format,

  - "SPRT" represents the short form for supportive courses
  - "MATH" represents the short form for mathematics courses
- 'XXX' represents a 3-digit numeric part of the course code with the following convention:
  - The first digit indicates the level of the course in terms of the year ('1' for 1st year, '2' for 2nd year, '3' for 3rd year, '4' for 4th year courses and '5' for 5th year courses);
  - The Second digit indicates level and similarity of the courses in the program
    - 0 designates foundation courses,
    - 1 designates support courses,
    - 2 designates requirements and software related courses,

- 3 designates programming courses,
- 4 designates database related courses
- 5 designates computer and network related courses,
- 6 designates mathematics and statistics and AI related courses,
- 7 designates management courses, 8 designates web related courses and 9 represents industrial capstone projects.

- The third digit indicates the semester within which the course is offered (odd numbers are given for courses given in the first semester and even numbers are given for courses given in the second semester)

For instance, SWEN322 means a software related course given for 3rd students in the 2nd semester.

2.10 Medium of Instruction

The medium of instruction for the program is ENGLISH

2.11 Graduation Requirements

2.11.1 Course Requirements

The overall student's workload in Credit hours is 179 with 18 credit hours per semester on the average (This means 304 European Credit Accumulation Transfer System (ECTS) with 30 ECTS/Semester on the average).

(i) Compulsory Courses (120 Cr.Hrs. – 203 ECTS):

Students must take and pass all of the following compulsory courses to graduate from the program.

Course Code	Course Title	Prerequisite	Cr. hours	ECTS
SWEN101	Introduction to Computer Systems	None	3	5
SWEN131	Fundamentals of Programming	None	3	5
SWEN104	Introduction to Software Engineering	SWEN101	3	5
SWEN132	Object Oriented Programming	SWEN131	4	7
ITSY154	Data Communication and Computer Networks I	SWEN101	3	5
ITSY256	Information Assurance and Systems Security	ITSY154	2	4
SWEN241	Fundamentals of Database Systems	SWEN101	3	5
SWEN223	Software Requirements Engineering	SWEN104	3	5
SWEN233	Data Structures and Algorithms	SWEN131	3	5

Course Code	Course Title	Prerequisite	Cr. hours	ECTS
SWEN224	Process Modeling and Workflow Design	SWEN223	3	5
SWEN232	Advanced Programming	SWEN132	4	7
SWEN226	Software Design and Architecture	SWEN223	4	7
SWEN252	Operating Systems	SWEN101	3	5
SWEN381	Web Systems and Services	None	3	5
SWEN331	Mobile Application Development	None	3	5
SWEN327	Enterprise Systems	None	3	5
ITSY363	Introduction to Artificial Intelligence	None	3	5
SWEN322	Software Quality Assurance and Testing	None	3	5
SWEN324	Software Usability and Management	None	3	5
SWEN376	Software Project Management	None	3	5
SWEN366	Methods for IS Research	MATH361	3	5
ITSY364	Foundations of Data Analytics	ITSY363	3	5
SWEN421	Software Process Improvement	None	3	5
SWEN423	Continuous Integration and Deployment	None	3	5
SWEN425	Service-oriented Architecture	SWEN327	3	5
SWEN471	Systems Thinking and Systems Approach	SWEN327	3	5
SWEN426	Seminar in Software Engineering	None	3	5
SWEN478	Software Product Management	None	3	5
SWEN492	Software Engineering Capstone Project I	None	4	7
SWEN522	Software Metrics	None	3	5
SWEN 524	Fundamentals of Financial Technology	None	3	5
SWEN552	Computer Simulation and Modelling	None	**3**	**5**
SWEN576	Management Information Systems	None	2	4
SWEN592	Software Engineering Capstone Project II	SWEN492	4	7
MATH161	Discrete Mathematics	None	3	5
MATH164	Linear Algebra	MATH161	3	5
MATH261	Calculus	MATH161	3	5
MATH266	Boolean Algebra	MATH164	3	5
MATH361	Statistical Methods	None	3	5
	Total Credit		120	203

(ii) Elective Courses (9 Cr Hrs. – 15 ECTS):

Students must take and pass a minimum of 9 credit hours of courses (15 ECTS) from the following list to graduate from the program.

Course Code	Course Title	**Prerequisite**	Credit hours	ECTS
SWEN437	Computer Graphics	None	3	5
SWEN439	Game Development	None	3	5
SWEN466	Knowledge Discovery and Data Mining	SWEN364	3	5
SWEN468	Software Agent	None	3	5
ITSY481	Cloud Computing and Data Centre Management	None	3	5
SWEN554	Ethical Computing	None	3	5

(iii) Support Courses (32 Cr. Hrs.– 56 ECTS)

Students must take and pass all of the following support courses to graduate from the program.

Course Code	**Course Title**	**Prerequisite**	**Credit hours**	**ECTS**
SPRT111	College English I	None	3	5
SPRT112	College English II	SPRT111	3	5
SPRT117	Introduction to Logic and Critical Thinking	None	3	5
SPRT115	Geography of Ethiopia and the Horn	None	3	5
SPRT118	Moral and Civic Education	None	2	4
SPRT214	Social Anthropology	None	2	4
SPRT217	General Psychology	None	3	5
SPRT311	Business Accounting and Management	None	3	5
SPRT312	Entrepreneurship	None	3	5
SPRT411	Inclusiveness	None	2	4
SPRT416	History of Ethiopia and the Horn	None	3	5
SPRT418	Basics of Organizational Behaviour	None	2	4
		Total Credit	32	56

2.11.2 Industry Practice (18 Cr. Hrs. – 30 ECTS)

As one of the critical components to enable a professional level work experience prior to graduation, a student is expected to join an organization as an intern and work for 18 hours a week for a minimum of 14 weeks, in order to have a real life industry experience and work environment. At the end of the industrial practice the student is required to write a detailed

summary of the experience gained. The paper should address the overall impression of the field in which the internship occurred, new skills acquired, contact made and how this experience may be helpful in the intern's future plans for graduate study or future employment.

Industry practice / Internship is a compulsory non-credit work and shall be recorded with a grade of "P" (Pass) and "F" (Fail), but neither shall be included in the computation of the Grade Point Average (GPA).

2.11.3 Community Service

In accordance with the community service guideline provided by the College, students are required to complete a minimum of 24 hours of community service in the course of their study.

The College shall provide a certificate of appreciation for the community service carried out by a student.

2.11.4 Cumulative Grade Requirements

To graduate from the program, students must pass every course taken during his/her stay in the College, Cumulative Grade Point Average (CGPA) of at least 2.0. A pass grade for a course is considered to be A, B+, B, C+, C or C-.

A student cannot graduate with a CGPA of less than 2.0. He shall thus score at least a `C` grade in each of the courses he is required to take under the program. However, a good standing student is entitled to graduate even if he scores a `D` grade in any course.

2.12 Degree Award

The degree awarded on successful completion of the undergraduate program in software engineering is ***"Bachelor of Science Degree in Software Engineering"***

2.13 Degree Nomenclature

English:

Bachelor of Science Degree in Software Engineering

Amharic:

የሳይንስ ባችለር ዲግሪ በ ሶፍትዌር ምህንድስና

2.14 Quality Assurance

The College shall ensure the quality of its undergraduate programs so as to achieve the objectives set for them and respond to the needs of students and society. Among the major activities to be carried out are:

- attracting qualified and committed staff;
- maintaining curricula that meet national and international standards;
- maintaining standard class sizes that allows close follow-up and individualized service
- Standardization of course offerings through preparation of general course outlines, exam contents, and external audit;
- the actual provision of opportunities for students to take what has been learnt in classroom and transform it into uses in the real world;
- use of state-of-the-art laboratories, computing facilities, and educational support materials;
- Periodical workshops (with stakeholders, teachers and graduates);
- Summative review of the program every five years
- Graduates' evaluation of the program;
- Assessments using survey project works/research, internships, and link programs;
- Annual assessment of the program;
- Establishing Alumni of Graduates as a mechanism to assess their career development.

3. Course Descriptions

3.1 Core/Compulsory Courses

3.1.1 SWEN101 Introduction to Computer Systems

Course Code:	SWEN101
Course Title:	Introduction to Computer Systems
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Lab Hours:	2 hours per week
Course Schedule:	Academic Year I
	Semester I
Description	This course provides a general introduction to computers, applications software, programming, hardware and computer information systems. Emphasis will be placed on modern computer system; procedural and assembly languages typically used for low-level programming of computer systems. Representation of data on computers. Comparisons of different types of instruction sets and corresponding addressing modes; relationships among instruction sets, fetch and execute operations, and the underlying architecture. Consideration of the physical implementation of large memory systems, together with the techniques of data storage and checking. Overall concepts of virtual memory, operating system functions, file systems and networks. Virtual machines and the levels of machine organization, the assembly and linking process and software libraries.
Learning Outcome	Upon completion of this course will be able to: ● Describe the layers of architectures in modern computer systems from hardware device levels upwards. ● Explain how the major components of a CPU work together, including how data is represented on a computer. ● Explain the basics of computer memories and their abstractions on modern computer systems. ● Design, implement and analyze programs at the machine code and assembly language levels, ● Construct small programs in the C programming language, and analyze their behavior. ● Describe the relationship between high-level procedural languages and assembly/machine language in the conventional machine layer, including how a compiled program is executed on a modern computer.

Course Content

Unit	Topic	Week
1	Modern Computer System's Architecture ● Introduction to computer systems (Computer Hardware, Computer Software & Liveware) ● Computer Architecture ● Von Numann Architecture ● Computer Organization ● Computer Architecture Design Goals	1-2
2	Data Representation in Computers ● Concepts of Data Representation in Digital Computers ● Binary Systems	3-5

	● Bits, Bytes, Nibble and Word ● Types of Data Representation ● Number Systems and Their Representation ● Binary Number System ● The Hexadecimal System	
3	. Logic Gates and Logic Circuits ● Logic Gates ● Functions of Logic Gates ● Logic Circuits	6-7
Mid Semester Week		8
4	Operating Systems ● Introduction to Operating Systems ● Processes and Threads ● Process synchronization ● Process Scheduling ● Main Memory Management ● Virtual Memory ● File Systems	9-12
5	Programs and Programming Languages ● Introduction to Programs ● Computer Program Design ● Introduction to Programming Languages ● Levels of Programming Language ● Procedural Language ● Introduction to programming in C ● Introduction to programming in Assembly Language ● Program Execution ● Interpreter ● Compilers	13-16

Textbook and References:	1. How to Design Programs: An Introduction to Programming and Computing (The MIT Press), 2018, by Matthias Felleisen and Robert Bruce Findler 2. Introduction to Computers and Information Technology 2nd edition by emergent learning (2019)
Particular resource req.:	Computer Lab,
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.2 SWEN131 Fundamentals of Programming

Course Code:	**SWEN131**
Course Title:	Fundamentals of Programming
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Lab Hours	2 hours per week
Course Schedule:	Academic Year I
	Semester I
Description	In this course the student will gain a broad understanding of modern computer programming. The student will acquire introductory skills in problem analysis, solution design, and program construction. Through practical programming activities, the student will gain an appreciation of the nature and history of computer programming. Introduction to computer programming. The main contents of the course are - Generations of computer language; Interpreted and compiled languages; Program design and development process; Problem definition; Pseudo-code; Flowcharting; Code modularization; Coding, testing, and debugging; Sequence, selection, and iteration patterns; Array processing; File operating, file input / output.
Learning Outcomes:	Upon successful completion of this course, the student will have reliably demonstrated the ability to: ● Demonstrate problem solving skills by developing algorithms to solve problems incorporating the concept of data abstraction in a computer program. ● use pseudo-code and visual modeling to prepare clear and accurate program documentation and models. ● examine working programs to identify their structures. ● Design programs according to specifications by creating flow charts, IPO charts and pseudo code. ● Implement a simple program by writing the code, testing the code and debugging the program. ● Incorporate the use of sequential, selection and repetition control structures into a program. ● Demonstrate an understanding of the design and implementation of functions ● Implement programs using sequential input and output files. ● Demonstrate an understanding of the use of the array data structure

Course Content

Unit	Topic	Week
1	Introduction ● The python programming language ● Installing Python ● Programming language and types ● Writing first python program ● Syntax and datatypes ● Variables ● Operators ● Arithmetic operators ● Logical operators ● Unary operators	1-2
2	Decision (branching) ● Introduction to conditional statements ● Simple if	3-4

	● if … else … ● if … else if … else … ● switch	
3	Repetitive Tasks ● Introduction to looping statements and flow control ● For loop ● While loop ● do...while loop	5-6
4	Functions ● Why functions? ● Passing arguments and returning value ● Keyword arguments ● Variable scope ● Default values ● Main function ● Recursive function	7
Mid Semester Week		8
5	Data structures ● List ● Tuple ● Dictionary ● Sets	9-12
6	File operations ● Opening a file ● Reading from file ● Writing to file ● Closing a file	13-16

Textbook and References:	1. Python Crash Course: A Hands-On, Project-Based Introduction to Programming, 2015, by Eric Matthes 2. Beginning Programming with Python For Dummies by john wiley (2023)
Particular resource req.:	Computer lab, Python, Sublime text, Visual Studio Code
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	Grade will be based both on concepts and practical applications. Examinations, written and lab assignments will be used to determine the grade. The evaluation shall be based on both formative and summative assessment which includes: 30%: Continuous Assessment, 20%: Project, 50%: Final Examination.

3.1.3 SWEN104 Introduction to Software Engineering

Course Code:	SWEN104
Course Title:	Introduction to Software Engineering
Prerequisites:	SWEN101
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year I Semester II
Description	This course covers the fundamental concepts and methodologies of software engineering. It emphasizes the main phases of the software lifecycle, such as requirements, design, implementation, testing, project planning. Also, it stresses the difference between the software product and process. It introduces concepts such as software processes and agile methods, and essential software development activities, from initial specification through to system maintenance. Formalisms and tools to assist in software development are also presented, including common design patterns and UML notation. There is a focus on software testing, from unit testing to the testing of software releases. Case studies provide practical examples for many of these concepts.
Learning Outcome	On successful completion of this course students will be able to: ● Explain the principles of software engineering with emphasis on the various phases of the software development life cycle ● Identify the issues relating to producing quality software ● Develop a system design using UML notation ● Explain human issues in the software engineering profession: ethics and professional practice ● Discuss the different aspects of project management in producing quality software

Course Content

Unit	Topic	Week
1	Introduction to software engineering ● Definition of software engineering (Difference with computer science) ● Categorization of software (characteristics, responsiveness, and type) ● Attributes of good software ● Software engineering methodologies ● Software engineering code of ethics in professional practice	1-2
2	Software development lifecycle ● Introduction to software development life cycle (SDLC) ● Activities and deliverables in a sequential life cycle model ● Activities and deliverables in an iterative life cycle model	3-4
3	Software modeling ● Introduction to UML artifacts ● Agile modeling concepts	5-6
4	Software requirements gathering ● Data types and data dimensions ● Data/requirements gathering techniques ● Data gathering techniques most appropriate for each application type ● Proposal and evaluation of proposal regarding hardware and software requirements	7
Mid Semester Week		8

5	Software requirements analysis ● Fundamental of software requirements and analysis ● Activities of software requirements and analysis ● Requirements elicitation techniques ● Data-oriented, process-oriented, and object-oriented methodologies ● Analysis activities and their major representations in data-oriented, process-oriented, and object-oriented methodologies.	9
6	Software design ● Software design principles ● Architectural design in terms of decisions, system organization, modular decomposition, and flow-and-control ● Design activities and their major representations in data-oriented, process-oriented, and object-oriented methodologies.	10-11
7	Implementation ● Programming introduction ● Characteristics and selection of programming/implementation languages ● Concepts for purchasing of hardware and software.	12-13
8	Software testing ● Basic software testing terminologies ● Testing strategies ● Design a test plan; unit, integration, and system levels test	14-15
9	Project management ● Role of the project manager relative to the software engineer ● Areas of responsibilities of a project manager ● Project management in terms of the project, people, and change management (i.e., planning, scheduling, execution, etc.)	16

Textbook and References:	1. Beginning Software Engineering Second edition by stephens, Rod (2023) 2. Introduction to Software Engineering, Second Edition by leach, ronald.j (2020)
Particular resource req.:	None
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.4 SWEN132 Object Oriented Programming

Course Code:	SWEN132
Course Title:	Object Oriented Programming
Prerequisites:	SWEN131
Credit Hours:	4 (7 ECTS)
Lab Hours:	2 hours per week
Course Schedule:	Academic Year I
	Semester II
Description	The course is designed to introduce students on how to develop business applications using object-oriented design methodology with Java as an illustration programming language. It includes Object Oriented Programming paradigm and its use; classes, Objects, Abstraction and Encapsulation; Inheritance, Polymorphism, Creating Graphical User interfaces (GUIs), Data Structures, Exceptions (Try, catch, and throw, finally how exceptions affect the design of an application)
Learning Outcomes:	At the end of the course students will be able to ● Understand major concepts of object-oriented programming ● understand the programming environment as defined by compilers, interpreters, editors, and other system software providing support for the programming activity ● develop skills in OO design and program development within an integrated development environment ● Use arrays and other data structures ● understand the concepts of encapsulation, inheritance ● Implement I/O functionality to read from and write to data and text files. ● understand object technology and its applications ● Explain the application of a variety of data structures and, understand the advantages and disadvantages of those structures

Course Content

Unit	Topic	Week
1	Introduction to Object-Oriented Programming (OOP) ● Overview of OOP ● Why Java? ● The JVM and Byte Code ● Basic concepts of OOP ● classes ● objects ● members ● class member visibility Encapsulation, inheritance and polymorphism	1
2	The inside of objects and classes: More on OOP concepts ● Member methods and their components ● Instantiation and initializing class objects ● Constructors ● default and parameterized ● overloaded constructors ● Methods ● Access specifiers ● Accessors and mutators ● Calling and returning methods ● Static and instance members	2-4
3	Inheritance ● Concept of inheritance ● Superclasses and subclasses	5-7

	● Protected members ● Overriding methods ● Using this () and super () ● Use of final with inheritance ● Constructors in subclasses	
Mid Semester Week		8
4	Polymorphism ● Introduction ● Relationships among objects in an inheritance hierarchy ● Assigning reference of subclass to superclass- ● type variable ● Assigning a superclass reference to subclass-type variable ● Subclass method calls via superclass-type variable ● Summary of allowed assignments between superclass and subclass variables ● Multiple inheritance and interfaces	9-11
5	Exception Handling ● Exception handling overview ● The causes of exceptions ● The Throwable class hierarchy ● Handling of an exception ● The throw statement ● The finally clause ● User defined exceptions	12-14
6	Files and Streams ● Introduction ● I/O classes ● File and FileDialog objects ● Low-Level File I/O ● High-Level File I/O ● Object I/O ● Random Access files	15-16
Textbook and References:	1. Java All-in-One For Dummies 7th edition by lowe (2023) 2. Learning java by Loy , Marc 6th edition revised(2023)	
Particular resource req.:	Computer lab, Java Development Environment (Eclipse / Netbeans), JDK, JRE	
Teaching Strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.	

3.1.5 ITSY154 Data Communication and Computer Networks I

Course Code:	**ITSY154**
Course Title:	**Data Communication and Computer Networks I**
Prerequisites:	SWEN101
Credit Hours:	3 (5 ECTS)
Lab Hours:	2 hours per week
Course Schedule:	Academic Year I Semester II
Description:	The course aims at exploring the various types of data communication systems, networks and their applications. The content includes: computer networks, seven-layer architecture, OSI & TCP/IP suite of protocols, network hardware, network software, standardization, guided transmission media, wireless transmission, switching and routing" data link layer, Ethernet and IP addressing. It involves practical session on Cabling and crimping, Configuring TCP/IP, Peer to Peer Networking, Sharing Files, Sharing Printers, Client-server Networking, Steps for Creating a home or small office Network, Experiencing collaboration tools, installing & Configuring Network Operating System, Exploring Server Roles, Setting up a DNS Server, setting up a DHCP server, Domain controller and IP Addressing, Basic concepts of wireless networking.
Learning Outcomes:	Upon successful completion of the course, students will be able to: ● Describe the basics of data communications, network and network equipment ● Explain the benefits and the need for network ● Understand data transmission and transmission media ● Understand Protocols and various networking components ● Understand TCP/IP & OSI Reference Model ● Demonstrate cable crimping, establishing, setup and troubleshooting Networks ● Understand basic concepts of addressing, switching and routing ● Understand network security and data integrity ● familiarize themselves with wireless networking

Course Content

Unit	Topic	Week
1	Introduction ● History & overview of Networks ● The impact of Networks on daily life ● The network as a platform ● Network Role & Elements ● Network Architecture Characteristics ● Computer Networks Versus Human Network	1-2
2	Data Communications ● What is communication? ● The platform for communication ● Data transmission ● Components of the network	3-4
3	Network Types ● LANs, WANs and Internetworks ● Peer to peer versus Server based Networks ● Packet-switched and Circuit switched networks ● Network cabling & Topologies	5

4	Protocols ● Rules & Network Protocols ● Protocol suites & Industry Standards ● Layered Models	6-7
Mid Semester Week		8
5	OSI Reference Model ● Layered Framework of OSI ● Overview & functions of each layer	9-10
6	Switching & Multiplexing ● Switching Concept and Types ● Multiplexing Concepts and Types ● Introduction to Ethernet & Wireless Networks	11-12
7	Introduction to IP Addressing and Subletting ● Classful & Classless Addressing ● Subletting and Variable Length Subnet Masking (VLSM)	13-14
8	Data Security and Integrity ● Fundamentals of secure networks; cryptography ● Encryption and privacy ● Authentication protocols ● Firewalls ● Virtual private networks ● Transport layer security	15-16

Textbook:	Computer Networking: Beginner's guide for Mastering Computer Networking and the OSI Model (Computer Networking Series Book 1), 2017 by Ramon Nastase
Textbook and References:	1. The software-defined network (SDN) by sahoo, kshira sagar (2023) 2. Computer networking 8th global edition by kurose, james (2021)
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessments which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.6 ITSY256 Information Assurance and Systems Security

Course Code:	ITSY256
Course Title:	Information Assurance and Systems Security
Prerequisites:	ITSY154
Credit Hours:	2 (4 ECTS)
Course Schedule:	Academic Year II
	Semester I
Description:	The course provides an introduction to information assurance. It covers fundamental concepts necessary to understand the threat to security as well as various defences against those threats. IT also examines fundamentals of network security involved in creating and managing secure computer network environments. Both hardware and software topics are considered, including authentication methods, remote access, network security architectures and devices, cryptography, forensics and disaster recovery plans. disaster recovery plans.
Learning Outcomes:	On successful completion of the course students will be able to: ● Define key terms and concepts of information assurance, ● Identify various threats, attacks and vulnerabilities to a computer system, ● Describe legal and ethical issues of information security, ● Identify various technical approaches to access control, intrusion detection and incident response ● Apply cryptography security technique, systems and Network security applications. ● understand how network security is conceptualized and carried out ● analyze both early and contemporary threats to network security ● familiarize themselves to concepts of cyber security and ethical hacking

Course Content

Unit	Topics	Week
1	Introduction ● Definition of Information Systems Security ● Critical concepts of Information Security ● Security/Privacy Vulnerabilities	1-2
2	Fundamentals of IS Security ● IS Security Fundamentals ● Components of Information Systems security ● Principles of Information Systems Security ● Introduction to IS Security Policy ● Planning, Design and Implementation of IS Security	3-5
3	Attack Types and Protection Schemes ● Categories of Attack Types and Security threats ● Vulnerabilities of Information Systems ● Malicious Security Threats ○ viruses ○ worms ○ Trojan horses ○ Spyware ● Categories of Security controls	6-7
Mid Semester Week		8

4	Security Techniques ● Cryptography ○ Introduction ○ Definitions and Terms ○ Private Key cryptosystems ○ Public key cryptosystems ○ Data Encryption Standards ○ Digital Signature ● Access Control ● Firewalls ● Intrusion Detection and Prevention Systems ● Authentication	9-11
5	Security at Different Layers ● Physical Security ● Software Security ● Network Security ● Web Security ● Advanced Security Issues	12-14
6	Risk Management ● Risk management strategies ● Disaster recovery plans	15-16
Textbook and References	1. Whitman, Michael and Mattford, Herbert (2015). Principles of Information Security (5th edition), Course Technology, Cengage Learning 2. Fundamentals of Information Systems Security, 2016, by David Kim and Michael G. Solomon	
Teaching Strategy	Instructor delivers lectures, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.	

3.1.7 SWEN241 Fundamentals of Database Systems

Course Code:	SWEN241
Course Title:	Fundamentals of Database Systems
Prerequisites:	SWEN101
Credit Hours:	3 (5 ECTS)
Lab Hours	2 hours per week
Course Schedule:	Academic Year II
	Semester I
Description:	The course covers the following topics: Database concepts related to data handling techniques, definition of a database and benefits of database systems, functions and components of DBMS. Architecture for database systems: ANSI SPRTARC architecture architectures, data model concepts and basic types of data models (Hierarchical, Network and Relational data models). Emphasize on Relational data model: data structures and integrity rules. Three levels Database design: (Conceptual, Logical and Physical Database designing). Basics of Relational Languages (Relational Algebra, Relational calculus and SQL), normalization as a process for verification of data model design, SQL interaction with programming interfaces.
Learning Outcomes:	At the end of the Course students should be able to: ● Explain what a Database System is, and be able to identify its characteristics and applications, ● Explain the different models of database, ● Design ER models from specifications and interpret them into relational tables, ● Write SQL statements for data creation and manipulation purposes, ● Describe how to optimize databases to the most efficient form, ● Distinguish and use relational model and relational algebra, ● Identify and fix the possible problems that may occur in securing data.

Course Content

Unit	Topic	Week
1	Introduction ● Data Handling approaches ● Roles in Database Design & Development o Data and Database Administrator o Database Designer o Application Programmer o End-Users ● The ANSI-SPRTARC Architecture ● Functions of DBMS ● Data models and conceptual models ● Database Languages (DDL, DML, DCL)	1-2
2	Relational Data Model ● Terminologies ● Relational Constraints ● Relational Integrity ● Key constraints ● Referential constraints ● Relational languages and views ● Relational DBMS	3-5

3	Conceptual Database Design- E-R Modeling ● Database Development Life Cycle ● Basic concepts of E-R model ● Entity type ● Attributes ● Relationship types ● Structural constraints ● Cardinality constraints ● Participation constraints ● Problem with E-R models ● Enhanced E-R models	6-7
Mid Semester Week		8
4	Logical Database Design ● Normalization ● Purpose of normalization ● Information redundancy and update anomalies ● Functional dependencies ● Process of normalization (1NF, 2NF, 3NF)	9-10
5	Physical Database Design ● Physical database design process ● Database design and implementation for relational databases	11-13
6	Query Languages ● Relational Algebra ● Relational calculus ● Structured Query Languages (SQL)	14-16

Textbook and References:	1. Database Systems: Design, Implementation, and Management, 2015, by Carlos Coronel and Steven Morris 2. Database Systems: A Pragmatic Approach, 3rd Edition by Foster , elvis (2022)
Particular resource req.:	Computer lab, SQL SWENRVER 2000 /My SQL/ PostgreSql
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.8 SWEN223 Software Requirements Engineering

Course Code:	SWEN223
Course Title:	Software Requirements Engineering
Prerequisites:	SWEN104
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year II
	Semester I
Description:	This course focuses on using a systematic approach to eliciting, analyzing, validating, documenting and managing requirements. It investigates the requirements engineering approach and the adoption of relevant techniques at each stage. The course starts with an overview of software requirements basics including definitions of terminology, describing software requirements, standards and an introduction to the requirements engineering process. UML tools and techniques are also covered. Theoretical concepts are introduced and are then reinforced through practical exercises and a running case study where students can apply techniques of analysis in a realistic project. The course aims to develop the necessary skills needed to work with requirements stakeholders and actors to make sure that requirements are complete, unambiguous, realistic and testable.
Learning Outcomes:	On successful completion of this course, students will be able to: ● Understand the Requirements Engineering purpose, approach and process ● Apply a range of requirements elicitation techniques ● Select the appropriate requirements elicitation techniques to identify requirements ● Model and document system and software requirements ● Analyze, prioritize and validate requirements ● Understand the principles and techniques of requirements management ● Understand principles to systematically establish, define, and manage requirements for a software; ● Use various requirement specification authoring techniques such as user stories and scenarios

Course Content

Unit	Topic	Week
1	Fundamentals of requirements engineering ● The essential software requirement ● Domain understanding ● Good practice for requirement engineering ● The business analyst role	1-2
2	Requirements development ● Establishing the business requirements ● User classes ● User personas ● Requirement elicitation ● Understanding user requirements ● Use cases and user stories ● Documenting requirements ● Specifying data requirements ● Software quality attributes ● Prototyping	3-7

	● Setting requirements priorities ● Requirements validation ● Requirements reuse ● Estimating requirements effort	
Mid Semester Week		8
3	Requirements for specific project classes ● Agile approach to requirements ● Adapting requirements practice to agile projects ● Requirement techniques when there is an existing system ● Packaged solution projects ● Outsourced projects ● Business process automation projects ● Business analytics projects	9-11
4	Requirements management ● Requirements management practice ● Change management practice ● Links in the requirements chain ● Tolls for requirement engineering	12-13
5	Implementing requirements engineering ● Improving requirements process ● The process improvement cycle ● Requirements engineering process assets ● Software requirements and risk management ● Software risk management ● Requirements-related risks	14-16

Textbook and References:	1. Sommerville, Ian (2017) Software Engineering, tenth edition, 2. Requirements Engineering for software and systems 4th edition by laplante, phillip A (2022)
Particular resource req.:	UML diagramming tools (Visio, Pencil)
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.9 SWEN232 Advanced Programming

Course Code:	SWEN232
Course Title:	Advanced Programming
Prerequisites:	SWEN132
Credit Hours:	4 (7 ECTS)
Lab Hours:	2 hours per week
Course Schedule:	Academic Year II
	Semester II
Description:	This course makes revision of software design and architecture and practical agile approaches to usable application software development, it then continues to familiarize students to DevOps and related workflows, Software design patterns, Software development workflow, tools and components, General setup and structure of software projects, Software versioning (version control), Software development and testing, Behavior driven development (BDD); User stories and scenarios, Writing / Generating tests from scenarios, Developing features (models, views, controllers) with Test-driven Development techniques; Software version control with Git.
Learning Outcomes:	At the end of the course, students will be able to ● Make practical and effective use of agile software development approaches and popular software design patterns ● Understand and get hands on experience on test driven software development, including automated testing techniques ● write better organized and testable code, produce quality and well tested software products which comply with basic testing standards and high-test coverage ● Make practical and effective use of programming tools

Course Content

Unit	Topic	Week
1	Introduction ● Installing rails ● Setting up development environment ● Choosing a rails version ● Rails and databases ● Creating a new application ● The architecture of rails application ● Models, views and controllers ● Rails model support	1-2
2	Introduction to ruby ● Data types ● Logic ● Functions ● Organizing structures	3
3	Building a ruby on rails application ● Incremental development ● Creating the application ● Validation and unit testing	4
4	Active record ● Defining data ● Locating and traversing records ● Creating, reading, updating and deleting (CRUD) ● Transactions	5-6

5	Action dispatch and controllers ● Dispatching request to controllers ● Processing of requests ● Objects and operations	7
Mid Semester Week		8
6	Action view and Migration ● Action View o Using Templates o Generating Forms o Processing Forms o Uploading Files to Rails Applications ● Migration	9-12
7	Customizing and extending rails ● Using Templates ● Generating Forms ● Processing Forms ● Uploading Files to Rails Applications	13-16

Textbook and References:	1. Practical Object-Oriented Design: An Agile Primer Using Ruby (2nd Edition), 2019 by Sandi MetzPractical Object-Oriented Design: An Agile Primer Using Ruby (2nd Edition), 2019 by Sandi Metz 2. https://guides.rubyonrails.org/getting_started.html
Particular resource req.:	Ruby, Ruby on Rails Framework, Rubymine community edition, Linux operating system preferred, cucumber, Rspec, Git
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which includes: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.10 SWEN233 Data Structures and Algorithms

Course Code:	SWEN233
Course Title:	Data Structures and Algorithms
Prerequisites:	SWEN131
Credit Hours:	3 (5 ECTS)
Lab hours	2 hours per week
Course Schedule:	Academic Year II
	Semester I
Description:	This course aims to introduce students some basic data structures and algorithms which are to be used as tools in designing solutions to problems. It will make students familiar with the specification, usage, implementation and analysis of these data structures and algorithms, Introduction of Data Structures (Linear, non-Linear Data Structures) mainly focused and Algorithm Analysis Concept, Measuring Complexity, Complexity of; Algorithm Big-O Notation. Simple Sorting and Searching Algorithms (Bubble Sort, Insertion Sort, Selection Sort, Sequential Searching, Binary Searching; Abstract Data Types, Structures, Pointers, Arrays, Linked Lists, Stacks, Queues, Trees, Graphs. Advanced Sorting and Searching Algorithms (Shell Sort, Quick Sort, heap Sort, Merge Sort, and Hashing); Laboratory exercises are dedicated to practice the basics on concepts on data structures like Abstract Data Types, Structures, Pointers, Arrays, Linked Lists
Learning Outcomes:	At the end of the course students will be able to ● Explain the basic techniques for the design and analysis of efficient Algorithm; ● Determine when and how to use the various data structures including Linked lists, Queues, Stacks, Binary trees, Search trees and Graphs; ● Design algorithms to solve real-life problems using the tools introduced; ● Analyze and efficiently implement solutions; ● Apply data structures and algorithms that are frequently used in information processing

Course Content

Unit	Topic	Week
1	Complexity analysis ● Computational and asymptotic complexity ● Big-O, Ω, Θ, little-o and OO notations ● Common complexity classes ● Best, average and worst-case complexity ● Amortized complexity	1-2
2	Linked lists ● Singly linked lists ● Doubly linked lists ● Circular lists ● Skip lists ● Self-organizing lists ● Sparse tables	3
3	Stacks and queues ● Stacks ● Queues ● Deques	4-5

	● Priority queue	
4	Recursion ● Recursive definitions ● Function calls and recursive implementation ● Tail recursion ● Nontail recursion ● Indirect recursion ● Nested recursion ● Excessive recursion ● Backtracking	6-7
Mid Semester Week		8
5	Simple Sorting and Searching Algorithms ● Searching Algorithm ● Linear Search (Sequential Search) ● Binary Search ● Sorting Algorithms ● Insertion Sort ● Selection Sort ● Bubble Sort ● Efficient sorting algorithms	9-10
6	Binary trees ● Trees, binary trees and binary search trees ● Implementing binary trees ● Searching a binary tree ● Tree traversal ● Breadth-first ● Depth-first ● Stackless depth-first ● Insertion , Deletion ● Balancing a tree ● Self-adjusting trees ● Heaps , Polish notation and expression trees	11-14
7	Hashing ● Hash functions ● Collision resolution	15-16
Textbook and References:	1. Data Structures and Algorithms Made Easy: Data Structures and Algorithmic Puzzles, Fifth Edition, 2017 by Narasimha Karumanchi 2. Introduction to algorithms and structures 1st edition by cengage (2023)	
Particular resource req.:	Python Programming Environment, Sublime Text, Visual Studio Code, etc.	
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.	

3.1.11 SWEN224 Process Modelling and Workflow Design

Course Code:	SWEN224
Course Title:	Process Modelling and Workflow Design
Prerequisites:	SWEN223
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year II
	Semester II
Description:	The aim of this course is to introduce students to the fundamental concepts of business process modeling, a systematic approach to model flow of work within organizations in order to support detailed analysis of business processes. The course covers definition of business processes, principles of process modeling, workflow design, analysis of business process models, overview of existing modeling languages (UML, YAWL, BPEL/BPMN), and business process integration. process representation; interpreting and creating process diagrams; and process validation and change management.
Learning Outcomes:	After completing this course, the student will be able to: ● use UML for modelling basic organizational and business processes, ● identify feedback dynamics in in organizational and business settings, ● develop cause-and-effect diagrams of problems for identifying major feedback loops, and simple models that can be simulated for analysis of organizational and managerial processes and problems. ● analyze a systemic problem that may impair the sustainable operation of an organization, ● develop strategies to solve the problem by making use of process modeling.

Course Content

Unit	Topic	Week
1	Introduction ● Learning from the past ● Process orientation ● Business process ● Business process approach	1-2
2	Establishment of process context, scope and goals ● Process discovery ● Steps in discovering business processes ● Establishment of process scope and content ● Steps in establishing process scope ● Process assessment ● Assessment by stakeholders ● Process differentiator ● Process enablers ● Environment in which process operators ● Measures ● Potential improvements	3-5

3	Understanding the As-Is process ● Process workflow models overview ● Essential elements of a Swimlane diagram ● Managing progressive details ● Process workflow models ● Development of As-Is process workflow model	6-7
Mid Semester Week		8
4	Design the to-be process ● Conducting a final process assessment ● Determining the to-be process characteristics and flow ● Process measurement ● Human resources ● Policies and rules ● Facilities design	9-12
5	Related requirement definition techniques ● Business-oriented data modelling ● Basic terms and concepts ● Business-oriented data modelling components ● Requirement modelling with use case and services ● From workflow to information system requirements ● Business services ● Use case concepts ● Methodology ● Service specifications ● Use case scenarios (conditions and outcomes) ● Complete use case scenario description (dialogues)	13-16

Textbook and References:	1. Workflow: A Practical Guide to the Creative Process, 2018, by Doron Meir 2. Workflow Scheduling on Computing Systems by Li, Kenli (2022)
Particular resource req.:	None
Teaching strategy:	Instructor delivers lectures, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.12 SWEN226 Software Design and Architecture

Course Code:	SWEN226
Course Title:	Software Design and Architecture
Prerequisites:	SWEN223
Credit Hours:	4 (7 ECTS)
Course Schedule:	Academic Year II
	Semester II
Description:	This course is designed to introduce students to the architecture and design of complete software systems, building on components and patterns. Topics to be covered include: Basic principles of Software Design, Moving from problem to solution, from what to how, from analysis to design, Problem Domain Modeling, Structure and Behavior Modeling, Class and Object Design, Software Architectures and Styles, Gang of four Design Patterns, GRASPRT Principles, Architecture and Design Refinement.
Learning Outcomes:	After completing this course, the student will be able to: ● Understand and apply various software design techniques ● Develop and evaluate software architectures ● Select and use appropriate architectural styles ● Select and use appropriate software design patterns ● Express the specifications and design of an application using UML, user stories, and scenarios ● Specify parts of the design using a formal design language ● Work effectively with a team of software project stakeholders, including customers and members of the development team.

Course Content

Unit	Topic	Week
1	Introduction ● Software architecture basic concepts ● Architectural structures and views ● Architectural patterns ● Importance of software architecture ● Contexts of software architecture ● Architecture in technical context ● Architecture in business context ● Architecture in professional context	1-3
2	Quality attributes ● Understanding quality attributes ● Availability ● Interoperability ● Modifiability ● Performance ● Security ● Testability ● Usability ● Other quality attributes ● Architectural tactics and patterns ● Quality attribute modelling ● Quality attribute analysis	4-7
Mid Semester Week		8
3	Architecture in software lifecycle ● Architecture in agile projects	9-11

	● Architecture and requirements ● Designing an architecture ● Documenting software architectures ● Architecture implementation and testing ● Architecture reconstruction and conformance ● Architecture evaluation ● Management and governance	
4	Architecture and business ● Economic analysis of architecture ● Architecture competence ● Architecture and software product lines	12-14
5	Architecture in the cloud ● Basic cloud definitions ● Service model and deployment ● Architecting in the cloud environment	15-16
Textbook and References:	1. Software Architecture with Python, 2017, by Anand Balachandran Pillai 2. Software Design by example by Wilson , Greg (2024)	
Particular resource req.:	Computer Lab	
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.	

3.1.13 SWEN252 Operating Systems

Course Code:	SWEN252
Course Title:	Operating Systems
Prerequisites:	SWEN101
Credit Hours:	3 (5 ECTS)
Lab hours	2 hours per week
Course Schedule:	Academic Year II
	Semester II
Description	This course examines basic issues in operating system design and implementation. It covers the tradeoffs that can be made between performance and functionality during the design and implementation of an operating system. Particular emphasis will be given to the major OS subsystems: process management (processes, threads, CPU scheduling), Memory management, file and I/O device management and deadlock), memory management (segmentation, paging, swapping) and file systems.
Learning Outcomes:	Upon the successful completion of the course students should be able to: ● Explain the objectives and functions of modern operating systems. ● Describe how operating systems have evolved over time from primitive batch systems to sophisticated multiuser systems. ● Analyze the tradeoffs inherent in operating system design. ● Describe the functions of a contemporary operating system with respect to convenience, efficiency, and the ability to evolve. ● Identify potential threats to operating systems and the security features design to guard against them. ● Describe how issues such as open source software and the increased use of the Internet are influencing operating system design.

Course Content

Unit	Topic	Week
1	Overview ● Role and purpose of operating systems ● history of operating system development ● Functionality of a typical operating system ● Design issues (efficiency, robustness, flexibility, portability, security, compatibility)	1
2	Processes and Threads ● Introduction to Processes and Threads ● Interposes Communication (IPC) ● Scheduling	2-3
3	Memory management (Main memory) ● Background ● Logical versus Physical Address Space ● Swapping ● Contiguous Allocation ● Paging ● Segmentation ● Segmentation with Paging ● Direct memory access	4-5
4	Processes Management ● Mutual exclusion: Definition of the “mutual exclusion” problem ● Deadlock detection and prevention ● Solution strategies	6-7

	● Models and mechanisms (semaphores, monitors, condition variables, rendezvous) ● Interrupt handling in a concurrent environment ● Producer-consumer problems ● Synchronization ● Multiprocessor issues	
Mid Semester Week		8
5	CPU Scheduling ● Pre-emptive and non-pre-emptive scheduling ● Scheduling policies ● Processes and threads ● Realtime issues	9-10
6	Device management ● Characteristics of serial and parallel devices ● Abstracting device differences ● Buffering strategies ● Recovery from failures	11-12
7	File System ● File systems: Fundamental concepts (data, metadata, operations, organization, buffering, sequential vs. nonsequential files); ● Content and structure of directories ● File system techniques (partitioning, mounting and unmounting, virtual file systems) ● Memory-mapped files ● Special-purpose file systems ● Naming, searching, and access ● Backup strategies	13-14
8	Security and protection ● Overview of system security ● Policy/mechanism separation; security methods and devices; protection, access, and authentication; models of protection ● Memory protection and Encryption ● Recovery management	15-16

Textbook and References:	1. Modern Operating Systems, 2016, by Tanenbaum Bos 2. Operating Systems: An Introduction, 2017, by R. Garg and G. Verma
Particular res. req.:	Linux, windows
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.14 SWEN381 Web Systems and Services

Course Code:	SWEN381
Course Title:	Web Systems and Services
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Lab hours	2 hours per week
Course Schedule:	Academic Year III
	Semester I
Description:	The objective of this course is to discuss how the Web systems are programmed and maintained and how online pages are created and delivered by Web servers and used by clients. Topics to be covered include: Web systems and technologies, information architecture, digital media, Web development, Web standards, vulnerabilities, social network software, client-side programming, server-side programming, Web services and servers, XHTML, CSS, CSS, Web systems security, JavaScript, PHP, and emerging technologies
Learning Outcomes:	At the end of the course, students will be able to ● Describe the core architecture of WWW as interconnected hypertext documents, the importance of Web protocols (e.g., HTTP), and the syntax and semantics of HTML, XHTML, XML, and CSS. ● Program Web applications using HTML, CSS, JavaScript and PHP. ● Implement client-side and server-side security methods for security and privacy. ● discuss how to organize information, build a website, and select graphical images, multimedia, and the use proprietary media and interaction technologies such as Flash, ActiveX, and QuickTime. ● Install, operate, and administer Web servers, proxies and caches.

Course Content

Unit	Topic	Week
1	Introduction to the Web ● Basics of Web services ● Web programming concepts ● Workflow Languages	1-2
2	Server-Side Scripting Basic ● Introduction to server-side scripting ● Server-side scripting languages ● Use Basic Syntax ● Send Data to the Web Browser ● Write Comments ● Utilize Variables ● Manipulate Strings ● Manipulate Numbers ● Work with constants	3-4
3	HTML Forms and Server-Side Scripting ● Use Conditionals and Operators ● Validate Form Data ● Send Values to a Script Manually ● Work with Forms and arrays of data ● Use for and While Loops ● Create a Simple Form using PHP	5-6

	● Receive Data from a Form in PHP ● Introduction to regular expressions	
4	Files and Directories ● Write to Files ● Read from Files ● Create Directories ● Upload Files ● Rename and Delete Files and Directories	7
Mid Semester Week		8
5	Connecting to Databases ● Connect to an existing Database ● Send Data to a Database ● Retrieve Data from a Database ● Modify Existing Data ● Remove Existing Data ● Data base security using server-side scripting	9-11
6	Cookies and Sessions ● Describe the stateless model ● Explain the concepts of maintaining state with sessions ● Create and Read data from sessions ● Putting PHP session IDs in pages ● Create and Read data from Cookies ● Destroy a session ● Maintain session data using Cookies ● Add Parameters to a Cookie ● Delete a Cookie	12-13
7	Content Management Systems (CMS) ● Concepts of CMS ● CMS development platforms	14-16

Textbook and References:	1. Web accessibility cookbook by Matuzovic , Manuel (2024) 2. Web Programming and Internet Technologies: An E-Commerce Approach, 2018, by Porter Scobey and Pawan Lingras
Particular resource req.:	Sufficient networked workstations with Apache/ Nginx web servers configured to run PHP.
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.15 SWEN331 Mobile Application Development

Course Code:	SWEN331
Course Title:	Mobile Application Development
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Lab hours	2 hours per week
Course Schedule:	Academic Year III
	Semester I
Description:	This course introduces students to programming technologies, design and development related to mobile applications. Topics include accessing device capabilities, industry standards, operating systems, and programming for mobile applications using the programming language discussed in class.
Learning Outcomes:	Upon completion, students should be able to ● devise and carry out test strategies of mobile design; ● create basic applications for mobile devices. ● debug a mobile application ● test a mobile application ● implement and evaluate techniques for the installation of mobile applications and delivery via various channels;

Course Content

Unit	Topic	Week
	The mobile ecosystem ● Devices ● Platforms ● Operating systems ● Application frameworks ● Mobile applications ● Services ● Size and scope of the mobile market ● The addressable mobile market ● Developing a mobile strategy ● Types of mobile applications	1-3
	Mobile design ● Interpreting design ● The mobile design Tent-Pole ● Designing for best possible experience ● The elements of mobile design ● Mobile design tools ● Designing for the right device ● Designing for different screen sizes	4-5
	Mobile web development ● Web standards and services ● Choosing mobile web options ● mobile web apps with HTML5 ● Adapting to devices	6-7
Mid Semester Week		8
	Mobile user interface design ● Effective use of screen real estate ● Understanding mobile application users ● Understanding mobile information design ● Understanding mobile platforms	9-12

	● Tools for mobile interface design	
	Android app development ● Android development tools ● Connecting to the google play ● Android development practice ● Building apps in android ● Common interactions ● Offline storage ● Web services ● GPS ● Accelerometer ● Testing an android application	13-16
Textbook and References:	1. Designing and developing innovative mobile applications by samanta , debabrata (2023) There will also be more references / Textbooks, based on the specific programming language the instructor uses to teach the course.	
Particular resource req.:	Mobile application development programming Environment per instructor's request	
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.	

3.1.16 SWEN327 Enterprise Systems

Course Code:	SWEN327
Course Title:	Enterprise Systems
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year III
	Semester I
Description:	The course covers the following Description: Enterprise-level information systems, technologies, and infrastructures; enterprise architecture and service management framework; business architecture (strategies, processes and governance), information architecture, applications architecture and technology architecture; enterprise information system design strategies, models and tools; evaluation of vendor strategies; legacy system migration issues, performance, interoperability, scalability, and security concerns; managing web-based client/server and distributed environments; web services foundations, vendor architectures, distributed applications; the context for integration, service-oriented application integration, multi-enterprise portals, mobile devices, business process integration; web design technologies, web services APIs, and emerging standards; implementation of enterprise resource planning package.
Learning Outcomes:	On successful completion of this course students will be able to: ● explain how information systems can transform organizations; ● analyze the role played by major types of information systems in organizations; ● analyze ethical and social concerns raised by enterprise networking; ● demonstrate how intranet and Internet technology can be used for e-business and e-operations; ● appraise system-building alternatives; ● select appropriate strategies to design and implement information systems.

Course Content

Unit	Topic	Week
1	Introduction ● Business processes and business process integration ● Acquiring and implementing enterprise systems	1-3
2	Analysis of Business Requirements ● Analyzing business requirements for selecting and implementing an enterprise system ● Selection of enterprise systems software ● Challenges associated with the implementation of global enterprise systems applications	4-7
Mid Semester Week		8
3	Organizational change and change management ● Strategic alignment ● User commitment ● Communications ● Training ● Job redesign ● Governance of processes and data	9-11

4	Business Process Implementation ● Post-implementation issues ● Enterprise system processes ● Order processing ● Purchasing ● Production logistics ● Accounting ● Planning and control	12-13
5	Human Resources ● Human resource functions ● How enterprise systems support business	14-16

Textbook:	The Practice of System and Network Administration: Volume 1: DevOps and other Best Practices for Enterprise IT (3rd Edition), 2016, by Thomas A. Limoncelli and Christina J. Hogan
Textbook and References:	1. Designing enterprise information systems by shishkov, boris (2020) 2. Software engineering for enterprise system agility by zykov, sergey (2018)
Teaching Strategy	Instructor delivers lectures, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessments which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.17 SWEN363 Introduction to Artificial Intelligence

Course Code:	SWEN363
Course Title:	Introduction to Artificial Intelligence
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Lab hours	2 hours per week
Course Schedule:	Academic Year III
	Semester I
Description:	The course explores basic principles, methodologies, techniques, tools and current research topics of Artificial Intelligence. The content includes: history and perspectives of AI, the different types of intelligent agents, goal based agents, search problems, constraint satisfaction problems, adversarial search problems, knowledge based agents, knowledge representation, inference techniques, propositional logic, first order logic, learning agents, inductive learning, neural networks, fuzzy logic, communication and perception, natural language processing, machine learning, computer vision and robotics. Application of these methods to important areas of Artificial Intelligence including development of knowledge-based systems.
Learning Outcomes:	On successful completion of the course students will be able to: ● Explain the different perspectives and historical background of Artificial Intelligence ● Describe different types and characteristics of intelligent agents ● Differentiate the different types of searching strategies employed in goal-based agents ● Represent knowledge and implement inference techniques ● Use learning algorithms to create decision tree ● Explain and demonstrate the use of neural network in implementing learning agents

Course Content

Unit	Topics	Week
1	Introduction to Artificial Intelligence (AI) ● Introduction to AI ● The Foundations of AI ● History of AI ● Approaches to AI ● State of the Art	1-2
2	Intelligent Agents ● Agents and Environments ● Rationality Vs Omniscience ● Structure of Intelligent Agents ● Agent Types o Simple reflex agent o Model-based reflex agent o Goal-based agent o Utility-based agent o Learning agent	3-4
3	Problem Solving (Goal Based) Agents ● Problem Solving by Searching ● Problem Formulation ● Search Strategies o Informed Search Strategies	5-7

	○ Uninformed Search Strategies ○ Local Search Strategies ○ Adversarial Search Strategies ● Avoiding Repeated States ● Constraint Satisfaction Search	
Mid Semester Exam		8
4	Knowledge Based Agents ● Logical Agents ● Propositional Logic ● Inference in Propositional Logic ● Predicate (First-Order) Logic ● Inference in First-Order Logic ● Knowledge Representation ● Knowledge-based Systems	9-12
5	Learning Agents ● Factors for designing learning agents ● Learning from Examples/Observation ● Knowledge in Learning ● Neural Networks	13-16

Textbook and References:	1. Artificial Intelligence and Machine Learning for Business: A No-Nonsense Guide to Data Driven Technologies, 2018, by Steven Finlay 2. Explainable agency in artificial intelligence by tulli, silvia (2024)C. Jackson,
Particular Resource Req.:	Computer lab, PROLOG, LISPRT or PYTHON
Teaching Strategy	Instructor delivers lectures, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.18 SWEN322 Software Quality Assurance and Testing

Course Code:	SWEN322
Course Title:	Software Quality Assurance and Testing
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year III
	Semester II
Description:	This course provides an introduction to software quality assurance concepts and testing. Quality assurance is viewed as an activity that runs through the entire development process: understanding the needs of clients and users; analyzing and documenting requirements; verifying and validating solutions through testing. Major topics are: Why do we do software testing? The meaning of black-box testing and white-box testing; Software Testing throughout the Software Process; Software Testing and Extreme Programming; The Automation of Software Testing; Difficulties and Limitations of Software Testing; The Business of Software Testing; Implementing and Automated Testing. Reasons for SQA failures and factors critical to success of SQA in IS development
Learning Outcomes:	At the end of the course, students should be able to ● Prepare a software quality plan for a software project ● Understand the methods and technologies of software testing; ● Design test plan and test cases; ● Designing tests that spot numerous ordinarily-overlooked defects in less time. ● Clearly and correctly report the software defectives; ● Asses the software product correctly; ● Distinguish relationship between the software testing and the quality assurance.

Course Content

Unit	Topic	Week
1	Introduction ● Defining software quality ● Software errors, defects, and failures ● Problems with defining requirements ● Software quality ● Software quality assurance	1-2
2	Why do we test software? ● When does a software go bad? ● Goals of testing software ● Components of a test plan	3

3	Test Design Concepts
	● Software Testing Foundations
	● Software Testing Activities
	● Testing Levels Based on Software Activity
	● Coverage Criteria
	● Test Design
	  ● Test Automation
	  ● Test Execution
	  ● Test Evaluation
	  ● Test Personnel and Abstraction
	● Pass/fail criteria
		4-5

4	Test automation ● Software testability ● Components of a test case ● Test automation framework ● Unit test frameworks ● Data-driven tests ● Beyond test automation	6-7
Mid Semester Week		8
5	Putting testing first ● Taming the cost-of-change curve ● Continuous integration ● System tests in agile methods ● Adding tests to legacy systems	9-10
6	Managing the test process ● Overview ● Requirements analysis and specification ● System and software design ● Intermediate design ● Detailed design ● Implementation ● Integration ● System deployment ● Operation and maintenance ● Implementing the test process	11-12
7	Writing effective test oracles ● What should be checked? ● Determining correct values ● Specification-based direct verification of outputs ● Redundant computations ● Consistency checks ● Metamorphic testing	13-14
8	Regression Testing for Evolving Software ● Regression testing design ● Regression testing implementation	15-16

Textbook and References:	1. Software Quality Assurance, 2018, by Claude Y. Laporte and Alain April 2. Introduction to Software Testing, 2017by P. Ammann and J. Offutt.
Particular resource req.:	None
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.19 SWEN324 Software Usability and Management

Course Code:	SWEN324
Course Title:	Software Usability and Management
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year III Semester II
Description	The class will cover the perceptual psychological, cognitive psychological, and other scientific underpinnings of usability (i.e., the emerging "usability science"), the usability engineering methods used in the pursuit of UCD, and the justification for the application of usability engineering in a software development
Learning Outcomes:	After completing this course, the student will be able to: ● understand and be able to explain the rudimentary aspects of how human beings take in and process information, ● know what the methods of usability engineering are and have experience with some of them, ● understand and be able to explain why software developers should NOT depend on their own intuitions for what is a usable design, ● be able to make the arguments for cost-justifying a user-centered design approach, ● have had exposure to a variety of usability labs, ● know how to carry out a usability evaluation and write a usability test plan and report.

Course Content

Unit	Topic	Week
1	Introduction ● Concepts of Usability ● Usability Engineering ● Attributes of Good software ● Case studies	1
2	User Interface Design: ● Importance of user interface design ● interaction styles ● Prototyping and iterative design, ● prototyping and defect correction, ● Participatory design approaches	2-3
3	Expert based Usability Inspection ● Heuristic evaluation method ● Cognitive Walkthrough method	4
4	User-Based Evaluation methods; ● Preparing interviews ● Preparing tests / creating scripts ● Real customers vs. potential customers ● Post-test question and answers ● Common pitfalls in software usability testing	5-7
Mid Semester Week		8
5	Setting up a usability test ● Testing Desktop application ● Testing mobile application	9

	● Live usability testing and recording	
6	Practical hands on user interface design and testing ● Low fidelity prototypes ● High Fidelity prototypes	10-12
Software Usability Testing Week		13-14
7	Software Usability Reporting ● Documenting software evaluation reports ● Presentation techniques	15-16

Textbook and References:	1. Cognitive systems engineering for user-computer interface design, prototyping and evaluation. By andriole, Stephen (2023) 2. Handbook of usability and user-experience by soares, marcelo M (2022) There will also be supplemental readings beyond the References Textbooks, such as articles or web pages, which will be assigned by the instructor throughout the semester.
Particular resource req.:	Usability Lab, Morae Tech Smith Usability Recording Software, Camtesia, screen recording software
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.20 SWEN376 Software Project Management

Course Code:	SWEN376
Course Title:	Software Project Management
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year III
	Semester II
Description:	The course covers topics such as IS plans and projects; practical examination of how software projects can be managed from start to finish; stages of project planning and project life-cycle; project selection from an organizational perspective, project initiation and scope development; team building and leadership; project costing, scheduling, and identifying and managing risks; product quality assurance techniques, project resource identification and allocation; project contracts management; progress and performance measurement and evaluation, project audit and closure; automated project management tools; ethical issues in software project management.
Learning Outcomes:	On successful completion of this course, students will be able to ● Define project management terms and techniques ● Evaluate and select projects ● Become familiar with project cost estimation and scheduling techniques and models ● Identify important risks facing a new software project ● Apply appropriate techniques to assess ongoing software project performance ● Explain and discuss the phases and knowledge framework for the methods used in software project management ● Explain the genesis of project, program, and portfolio management and their important to organizations' success. ● Apply project management process concepts by working on a team project as project manager or active team member.

Course Content

Unit	Topic	Week
1	Introduction to software Project Management ● Projects and Project Management ● Project Life Cycle Models and Paradigms	1
2	Software Project Scope Management & Planning ● Project Planning ● Project Scope Management ● Project Time Management ● Project Cost Management ● Project Risk Management	2-5
3	Project Organization ● Project Roles and Team Organization ● Staffing the Project ● Training ● Project Communication	6-7
Mid Semester Week		8
4	Productivity and Quality ● Measurement ● Quality Assurance	9-12

5	Remnants ● Project Procurement Management ● Project performance measure and evaluation ● Post-Project audits ● Ethical issues in project management	13-16
Textbook and References:	1. Project Management College (2017) A Guide to the Project Management Body of Knowledge: PMBOK(R) Guide 2. The performance of projects and project management by lecoeuvre , Laurence (2024)	
Particular resource req.:	Project management software tools	
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.	

3.1.21 SWEN366 Methods for IS Research

Course Code:	SWEN366
Course Title:	Methods for IS Research
Prerequisites:	MATH361
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year III
Description:	Semester II This course enables students to understand concepts and application of research. It attempts to define what research is, why they do research, and the various methods that researchers use to investigate problems. It is designed as an under-graduate introduction to research methodology in software engineering and information systems. The course provides a framework for conceptualizing research and is meant to underpin the research project for the final year. Special focus will be made in Design Science Research
Learning Outcomes:	At the end of the course students are expected to: ● Explain the purpose of research and understand basic concepts of research ● Understand research as a scientific method to acquire knowledge ● Acquire skill to formulate researchable problems ● Identify major research problems in Information Systems and related areas ● Explain the different research approaches and methods (Quantitative, qualitative and Design science) ● Be able to design research projects or research proposal ● Be able to collect and analyze data relevant to the research problem

Course Content

Unit	Topics	Week
1	Overview of research ● Definitions and conceptualization ● Characteristics of research ● Motivation and significance of research ● Types and purposes of research ● Theory and hypothesis in research ● Research Vs. Project	1-2
2	The Research Process ● Selecting a topic ● Formulating the Research Problem ● Literature Review ● Hypothesis Development ● Research Design ● Data collection and analysis ● Research report and communication	3-4
3	Research Design ● Important concepts in research design ● Sampling and sampling techniques ● Data collection techniques ● Data organization and analysis	5-7
Mid Semester Week		8
5	Scientific Communication / Research writing ● Research proposal writing	9-10

	● Research report writing ● Writing a journal article ● Scholarly communication and presentation skills	
	Ethics in Research ● Ethical Concerns ● Ethical issues in Data Collection ● Ethical issues in data analysis Ethical issues in Writing a research report	11-12
6	Design Science Research ● Placing Design Science Research in Context. ● Difference between routine design practice and design science research. ● Design Science Research Process	13-14
Assignment presentation and reflections		15-16

Text Book and References:	1. Williamson, Kirsty and Johanson, Graeme (2018) Research Methods: Information, Systems and Contexts 2. Handbook of qualitative research methods for information systems by Davison, Robert M.(2023)
Particular resource req.:	SPSS statistical package software, Online data Collection tools.
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.22 ITSY364 Foundations of Data Analytics

Course Code:	ITSY364
Course Title:	Foundations of Data Analytics
Prerequisites:	MATH361
Credit Hours:	3 (5 ECTS)
Lab Hours:	2
Course Schedule:	Academic Year IV
	Semester I
Description:	This course is intended to help students build a foundation for data science and data analytics. It introduces the basic concepts of data science laying emphasis on data visualization and statistical techniques to explore complex data. Further, the course will familiarize students with large datasets, including data, cleaning, model building and interpretations of experimental results. The course also introduces students with the concepts of big data and data warehousing.
Learning Outcomes:	By the end of the course, students will be able to: ● describe what Data Science is and the skill sets needed to be a data scientist, ● explain in basic terms what Statistical Inference means, identify probability distributions commonly used as foundations for statistical modeling, fit a model to data, ● use R or Python to carry out basic statistical modeling and analysis, ● explain the significance of Exploratory Data Analysis (EDA) in Data Science, ● apply basic tools (plots, graphs, summary statistics) to carry out EDA, ● describe the Data Science Process and how its components interact, ● use APIs and other tools to scrape the Web and collect data, ● apply EDA and the Data Science process in a case study ● understand the techniques and principles of big data and data warehouse

Unit	Topic	Week
1	Concepts of Data warehousing ● Introduction to Data Warehousing ● Big Data Analytics and Benefits ● Data Mining Concepts	1-2
2	Introduction to Data Analytics ● Overview of Data Analytics ● Importance and Applications ● Data Analytics Lifecycle	3-4
3	Data Collection and Preprocessing ● Data Types and Sources ● Data Collection Techniques ● Data Cleaning and Transformation ● Handling Missing Data ● Data Normalization and Standardization	5-6
4	Exploratory Data Analysis (EDA) ● Descriptive Statistics ● Data Visualization Techniques ● Identifying Patterns and Trends ● Correlation and Causation	7
Mid Semester Week		8

5	Introduction to Machine Learning ● Overview of Machine Learning ● Supervised vs. Unsupervised Learning ● Key Algorithms (e.g., Regression, Classification, Clustering) ● Model Evaluation and Validation	9-12
6	Data Analytics Tools and Technologies ● Introduction to Python/R for Data Analytics ● Libraries and Packages (e.g., Pandas, NumPy, Scikit-Learn) ● Building Shiny apps	13-16
Textbook and References:	1. Data Analytics: A Practical Guide To Data Analytics For Business, Beginner To Expert(Data Analytics, Prescriptive Analytics, Statistics, Big Data, Intelligence, Master Data, Data Science, Data Mining), 2017, by James Fahl 2. Big data analytics: A guide to data science practitioners marking the transition to big data. By Ulrich Matter (2023)	
Particular Resource Req.:	R programming environment, Python development environment	
Teaching Strategy:	Instructor delivers lectures, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination	

3.1.23 SWEN421 Software Process Improvement

Course Code:	SWEN421
Course Title:	Software Process Improvement
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year IV
	Semester I
Description:	This course aims to introduce students to software process improvement. Process improvement aims to learn from current practice and objectively assess potential improvements. This will be explored by practicing a simplified form of the Personal Software Process and studying a number of process related topics drawn from: the goal question metric paradigm; appropriate automation; configuration management; project tracking and control; quality assurance; cost of quality; continuous integration; DevOps; software distribution; Infrastructure, Platform and Software as a Service; leveraging social media and the internet.
Learning Outcomes:	At the end of the course students should be able to, ● Explain the importance of software process improvements in delivering quality software. ● Adopt and adapt various software process improvement frameworks for their own uses ● articulate a critical view of software process improvement and its significance, ● articulate a critical view of the PSPRT, ● articulate a critical view of their own software development process, and ● apply a disciplined personal process to their own work.

Course Content

Unit	Topic	Week
1	Introduction ● Software process ● Software process improvement ● Process mapping ● Process improvement initiatives ● Challenges in software engineering ● Software process and lifecycle ● Software inspection ● Software testing	1-4
2	Capability maturity model integration (CMMI) ● Introduction to CMMI ● CMMI maturity levels ● CMMI processes	5-7
Mid Semester Week		8
3	Setting up a CMMI ● Approach to continuous improvement ● CMMI improvement structure and terms ● Planning improvement cycle ● Implementation of improvements ● Piloting process	9-12

4	CMMI implementation
	● Project management
	● Supplier agreement management
	● Requirements development and management
	  ● Process map
	  ● Requirements procedure
	  ● Requirements template
	  ● Requirements checklist
	● Configuration management
	● Process and product quality assurance
	● Measurement and analysis
		13-16

Textbook and References:	1. Software architecture metrucs : case studies to improve the quality of your architecture by Christian ciceri (2022) 2. Software engineering measurement by munson (2019)
Particular resource req.:	None
Teaching strategy:	Instructor delivers lectures, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.24 SWEN423 Continuous Integration and Deployment

Course Code:	SWEN423
Course Title:	Continuous Integration and Deployment
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Lab hours	2 hours per week
Course Schedule:	Academic Year IV
	Semester I
Description:	This course introduces the concept of practices of Continuous Integration and Deployment. The course will provide DevOps Fundamentals: principles and practices, Version Control Systems; Continuous Integration; Continuous; integration, build automation and languages dependency Automated Software Testing
Learning Outcomes:	At the end of the course, students will be able to: ● Appreciate the fundamentals of DevOps and apply its principles and practices to a software development project ● Design and implement a continuous integration pipeline for a software development project ● Design and implement a continuous delivery/deployment stage of pipeline for a software development project ● Include a set of automated tests

Course Content

Unit	Topic	Week
1	Introduction ● Agile and continuous delivery ● The principle of flow ● The principle of feedback ● The principles of continual learning and experimentation	1-2
2	Starting the DevOps process ● Selection of value streams ● Understanding the work ● Organization and architecture design ● Integrating operations into the daily work of development	3-4
3	The technical practice of flow ● Creating foundation of deployment pipeline ● Enabling fast and reliable automated testing ● Enabling continuous integration ● Automate and enable low-risk releases ● Architecture for low risk releases	5-7
Mid Semester Week		8
4	The technical practice of feedback ● Telemetry creation to enable seeing and solving problems ● Analyse telemetry to better anticipate problems and achieve goals ● Enabling feedback so development and operations can safely deploy code ● Integrate hypothesis-driven development ● Review and coordination process to increase quality of work	9-12

5	Continual learning and experimentation ● Enabling learning into daily work ● Converting local discoveries into global improvements ● Organizational learning and improvement	13-16
Textbook and References:	1. DevOps Handbook: Introduction to DevOps and its impact on Business Ecosystem, 2017, by Stephen Fleming 2. DevOps: Continuous Delivery, Integration, and Deployment with DevOps: Dive into the core DevOps strategies, 2018, by SricharanVadapalli	
Particular resource req.:	DevOps tools such as Jenkins and Docker	
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.	

3.1.25 SWEN425 Service Oriented Architecture

Course Code:	SWEN425
Course Title:	Service Oriented Architecture
Prerequisites:	SWEN327
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year IV
	Semester I
Description:	This course focuses on Service Oriented Architecture (SOA) concepts and principles, as well as quality considerations for developing modern software systems from a technical and organizational perspective. Topics include: Introducing service oriented architecture (SOA); Principles of service orientation; SOA business aspects: standards of Web services, implementation of SOA using Web services, business aspects of SOA and Web services; SOA Design Patterns: SOAP - Message exchange Patterns - Coordination - Atomic Transactions - Business activities - Orchestration - Choreography - Service layer abstraction - Application Service Layer - Business Service Layer - Orchestration Service Layer; Business-centric SOA - service modeling - Service Oriented Design; SOA Technologies - SOA Tooling - SOA Vendors;
Learning Outcomes:	At the end of the course students will be able to: ● explain the meaning of the "Service Oriented" paradigm both from the business and technical point of view; ● understand the applicability of SOA design patterns and the meaning of the major SOA implementation technologies; ● compare SOA with other architectural paradigms; ● analyze requirements towards the creation of a service; ● design a service starting from the analysis phase; ● understand the problems in service design and analysis; ● understand the challenges in service implementation; ● being able to classify and make reasoned decision about the adoption of different SOA platforms;

Course Content

Unit	Topic	Week
1	SOA and web services fundamentals ● Introducing SOA ● The evolution of SOA ● Web services and primitive SOA	1-2
2	SOA and web services ● Activity management and composition ● Service activity ● Coordination ● Atomic transaction ● Orchestration ● Choreography ● Advanced messaging, metadata and security ● Addressing ● Reliable messaging	3-5
3	SOA and service orientation ● Principles of service orientation ● Service layers ● Service layer abstraction ● Application service layer	6-7

	● Business service layer	
Mid Semester Week		8

4	Building SOA
	● SOA delivery strategies
	  ● Top-down strategy
	  ● Bottom-up strategy
	  ● Agile strategy
	● Service oriented analysis
	● Service modelling
	● Service modelling guidelines
		9-12

5	Building SOA, technology and design ● Service oriented design ● SOA composition ● Core SOA standards ● Service design ● Application service design ● Business process design ● Fundamental web service extensions ● SOA platforms	13-16
Textbook and References:	1. Service-Oriented Architecture: Analysis and Design for Services and Microservices (2nd Edition), 2016, by Thomas Erl	
Particular resource req.:	None	
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.	

3.1.26 SWEN471 Systems Thinking and Systems Approach

Course Code:	SWEN471
Course Title:	Systems Thinking and Systems Approach
Prerequisites:	SWEN327
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year IV
	Semester I
Description:	This course focuses on approaches to systems thinking; systems-thinking method; and Systems Thinking Guide in the work place. Systems thinking as a method and tool for managing change, solving complex problems, and creating individual and team learning.
Learning Outcomes:	On successful completion of this course students will be able to: ● Gain an understanding of the language and concepts of systems, systems thinking, and complexity, and their implications for the workplace ● Gain an understanding of specific types of systems, that may be at play within complex problems ● Practice using a comprehensive Systems Thinking Guide to apply in understanding of systems thinking to a challenging situation and opportunity ● Develop an action plan to deal with the organizational problem and opportunity ● Gain an understanding of how to use systems thinking in a variety of situations

Course Content

Unit	Topic	Week
1	Systems thinking: general concepts ● Objects and events ● Structure, behaviour and discipline ● Matter, energy and information ● Historical background of system concept ● General system theory ● Systems thinking ● Human being as a complete and superior system	1-3
2	Systems and related concepts ● Different levels of systems concepts ● System environment concept ● Systems hierarchies ● Systems types, inputs and outputs ● Entropy and its concept in systems	4-7
Mid Semester Week		8
3	Systems structure, behavior and discipline ● System structure ● Systems behaviour ● Systems discipline ● Stability as structural balance ● Behavioural equilibrium ● Disciplinal certainty	9-12
4	Systems thinking ● Systems thinking concept ● Systems thinking methods and tools ● Systems description in ordinary language	13-16

	● Abstraction ● Modelling and simulation ● System diagrams ● Soft systems and hard systems	
Textbook and References:	1. Systems Thinking For Social Change: A Practical Guide to Solving Complex Problems, Avoiding Unintended Consequences, and Achieving Lasting Results, 2015, by David Peter Stroh 2. Gharakhani Bahar (2014) System and Systems Thinking: (Whole Review) 3. Jimmy Brown (2012) Systems Thinking Strategy: The New Way to Understand Your Business and Drive Performance 4. David Kerr (2012) An Introductory Guide to Systems Thinking 5. Jamshid Gharajedaghi (2011). Systems Thinking, Third Edition: Managing Chaos and Complexity: A Platform for Designing Business Architecture	
Particular resource req.:	None	
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.	

3.1.27 SWEN426 Seminar in Software Engineering

Course Code:	SWEN426
Course Title:	Seminar in Software Engineering
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year IV
	Semester II
Description:	The purpose of this course is to give students the opportunity to cover issues and current trends that might have not been covered in the courses provided as core or elective courses. The instructor has the responsibility of introducing current topics relevant for the program. Students are provided with a list of papers published on accredited journals or conference proceeding to choose from. Each student will choose papers, critically evaluate, prepare and submit a well-written report followed by oral presentation findings and critics.
Learning Outcomes:	On successful completion of this course, students will be able to: ● Get professional updates in the field of software engineering ● Hear state-of-the-art recommendations from expert faculty and guest lecturers on software engineering and related fields ● Recognize emerging technologies in software engineering and related fields.
Course Content	
Topics vary according to the interest of students and instructor. Typical topics include ● Latest research findings in software engineering ● Devops ● Block Chain ● Open Source Computing, etc.	
Textbook and References:	As suggested by respective instructors
Particular resource req.:	None
Teaching strategy:	Interactive discussions, student led presentations, case studies and real world applications, critical reading and analysis, guest lectures, discussion forums, reading assignments; workshop and evaluation etc.
Assessment:	Students will be assessed based on regular attendance, Active participation in discussions, contributions to discussions, and quality of questions and comments, reports from readings and case studies, written reflections. Marking and grading will be based on the types of assessment the instructor prefers to carry out.

3.1.28 SWEN478 Software Product Management

Course Code:	SWEN478
Course Title:	Software Product Management
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year IV
	Semester II
Description:	The course covers the entire software product life cycle, the emphasis is on requirements management and setting their priorities, feature grouping and variation management, and version control systems. The course will also cover issues related product management role in software industry; Product definition; configuration management; Product management tools - planning, managing and tracking and dealing with external stakeholders.
Learning Outcomes:	Upon successful completion, students will be able to: ● understand how software product management takes place in the scope of contemporary software development approaches. ● understand the value of process, requirements, planning, and monitoring in producing better software. ● relate software product management to better software products ● recognize the role of a software product manager ● reflect on how Agile principles will improve software projects

Course Content

Unit	Topic	Week
1	Introduction ● External and internal views on software product ● Customer-specific software product ● Product platform, family and line ● Product name version number and compatibility ● Attributes of software products	1-2
2	Software as a business ● Business aspects of software ● The financial life-cycle of a software product ● The software ecosystem ● Law of increasing returns ● Business model for software vendors ● Relationship between software product management and software pricing	3-4
3	Elements of software product management ● The role of software product manager ● Framework ● Market analysis ● Product analysis ● Product strategy ● Product planning ● Development ● Marketing ● Sales and distribution ● Support and services ● Tool support	5-7
Mid Semester Week		8
4	Elements of software pricing	9-12

	● Role of software pricing manager ● Software pricing framework ● Pricing strategy ● Price structure, policy and level ● Pricing in distribution channels ● Pricing for large customer accounts ● Negotiation ● Pricing in the global market ● Business-to-consumer (B2C) software ● Software as a service (SaaS) ● Pricing for corporate IT organizations	
5	Software product management and pricing in corporate structures ● Software product management in the internal environment ● Software pricing in the internal environment ● Organizational alternatives ● Scenarios	13-16

Textbook and References:	1. Software Product Management: The ISPRTMA-Compliant Study Guide and Handbook, 2017, by Hans-Bernd Kittlaus and Samuel A. Fricker 2. Product Management in Practice: A Real-World Guide to the Key Connective Role of the 21st Century, 2018, by Matt LeMay
Particular resource req.:	None
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.29 SWEN492 Software Engineering Capstone Project I

Course Code:	SWEN492
Course Title:	Software Engineering Capstone Project I
Prerequisites:	None
Credit Hours:	4 (7 ECTS)
Course Schedule:	Academic Year IV
	Semester II
Description:	The capstone project allows students to demonstrate their learning using an area of interest as the basis for the project. This could be in an area that they participate to pursue after graduation. Students will work in a team of 3-4 to design, assemble/develop and present a capstone project to an audience to demonstrate personal learning and achievement, and growth in core competencies
Learning Outcomes:	The following are the learning outcomes of the capstone project: ● Communication: In addition to written documentation of the project, students have the opportunity to develop their oral communication skills by way of providing presentations ● Lifelong learning: Students will perform independent learning of new technologies and concepts ● Modern Tools and Techniques: The completion of the project will enable students to select, and learn the necessary tools and techniques that are needed to complete the project.
Textbook and References:	Students will choose own reference materials based on the topic of their project.
Particular resource req.:	Computer Lab, students' choice of resources will be made available.
Teaching strategy:	Projects are carried out with continuous interaction between candidates and their designated supervisors.
Assessment:	The project is assessed through evaluation of the written report and the oral defense made by each candidate. An examination board set up for a project makes the assessment for each individual candidate.

3.1.30 SWEN522 Software Metrics

Course Code:	SWEN522
Course Title:	Software Metrics
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Lab Hours:	2 hours per week
Course Schedule:	Academic Year V
	Semester II
Description:	This course is a step by step description of the software metrics. It includes introduction to foundations of measurement theory, models of software engineering measurement, software products metrics, software process metrics and measuring management.
Learning Outcomes:	Upon successful completion, students will be able to: ● Understand the theoretical aspects of software measurements. ● Demonstrate the knowledge of software metrics. ● Demonstrate the knowledge of using software metrics in software development, software maintenance, and software project management. ● Demonstrate the knowledge of statistical analysis in software measurement. ● Demonstrate the knowledge of developing and calibrating predication systems. ● Demonstrate the knowledge of developing and maintaining a measurement program.

Course Content

Unit	Topic	Week
1	**Overview of software metrics** ● Basics of software measurement ● Software metrics Concepts	1
2	**The basics of measurement** ● Metrology ● Property-oriented measurement ● Meaningfulness in measurement ● Measurement quality and measurement process ● Measurement validation ● Object-oriented measurement ● Subject-domain-oriented measurement	2
3	**Goal-based framework for software measurement** ● Software measure classification ● Goal-based paradigms: Goal-Question-Metrics (GQM) and Goal-Question-Indicator-Metrics (GQIM) ● Applications of GQM and GQIM Case studies	3-4
4	**Empirical investigation** ● Software engineering investigation ● Investigation principles ● Investigation techniques ● Formal experiments: Planning ● Formal experiments: Principles ● Formal experiments: Selection ● Guidelines for empirical research	5
5	**Measuring internal product attributes: size**	6-7

	● Software size ● Software Size: Length (code, specification, design) ● Software Size: Reuse ● Software Size: Functionality (function point, feature point, object point, use-case point) ● Software Size: Complexity	
Mid Semester Week		8
6	**Measuring internal product attributes: structure** ● Software structural measurement ● Control-flow structure ● Cyclomatic complexity ● Data flow and data structure attributes ● Architectural measurement	9-10
7	**Measuring cost and effort** ● Software cost model ● Constraint model ● Software Lifecycle Management (SLIM) ● Cost models: advantages and drawbacks	11-12
8	**Measuring software quality reliability** ● Basic software quality metrics ● Quality management models ● Reliability concepts and definitions ● Software reliability models and metrics ● Fundamentals of software reliability engineering (SRE) ● Reliability management models	13-14
9	**Software test metrics** ● Test concepts, definitions and techniques ● Estimating number of test case ● Allocating test times ● Decisions based on testing ● Test coverage measurement ● Software testability measurement ● Remaining defCP measurement	15-16
Textbook and References:	1. A guide to selecting software measure and metrics by jones, capers (2023) 2. Software Architecture Metrics: Case Studies to Improve the Quality of Your Architecture, 2022. by Christian Ciceri et al.	
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.	

3.1.31 SWEN524 Fundamentals of Financial Technology

Course Code:	SWEN524
Course Title:	Fundamentals of Financial Technology
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Lab Hours	Academic Year V
Course Schedule:	Semester II
	SWEN524
Description:	This course is designed to provide students from non-finance backgrounds an in-depth understanding of both the opportunities and challenges in the FinTech industry, with a special focus on the Ethiopian market. The course will cover key FinTech concepts, digital currencies, digital payments, lending platforms, and regulatory landscape as well as delve into the critical areas of cybersecurity and fraud prevention in the FinTech sector. The course prepares students for potential careers in the rapidly evolving FinTech industry, with a strong foundation in both FinTech and cybersecurity principles.
Learning Outcomes:	At the end of the course students should be able to: ● have a comprehensive understanding of the FinTech industry and its significance in the Ethiopian market, ● understand key concepts and trends in FinTech, including digital currencies, digital payments, lending platforms, and the regulatory landscape. ● have an understanding of cybersecurity principles and their application in the FinTech industry. ● Recognize and understand various types of cyber threats and frauds prevalent in the FinTech sector ● Familiarize with the tools and techniques used for cybersecurity and fraud prevention in FinTech ● gain practical experience through case studies and group projects, allowing students to apply the concepts and techniques learned during the course

Course Content

Unit	Topic	Week
1	Introduction ● Understanding FinTech and its global significance ● Overview of the financial services industry in Ethiopia ● Digital payments and transactions in Ethiopia ● cryptocurrencies and their potential in Ethiopia ● Case Study: Bitcoin in Africa	1-2
2	Digital Payments, Transactions, Lending and Crowdfunding ● Exploring mobile money and banking in Ethiopia (M-BIRR, HelloCash) ● Introduction to Ethiopian payment gateways and POS systems ● Guest Lecture: from any/ of those: Telebirr, Amole Digital Wallet, Chapa, MPesa ● Case Studies ● Understanding online lending and crowdfunding ● Introduction to Ethiopian microfinance institutions and their potential digital transformation ● Case Studies	3-5

3	Regulatory Landscape and Future of FinTech in Ethiopia ● Regulatory environment for FinTech in Ethiopia ● Discussion of challenges and opportunities for FinTech in Ethiopia ● Exploring emerging trends and technologies in Ethiopian FinTech ● Understanding the potential impact of AI and machine learning on FinTech in Ethiopia ● Concluding discussion: The future of finance and technology in Ethiopia	6-7
Mid Semester Week		8
4	FinTech Project ● Group project: Develop a concept for a new FinTech service or product for the Ethiopian market ● Presentation of group projects	9-10
5	Cybersecurity, Fraud Detection and Prevention in FinTech ● Recognizing common types of cyber threats in FinTech ● Introduction to cybersecurity tools and measures ● Understanding the role of secure coding and software development in FinTech ● Understanding various types of fraud in the financial sector ● Introduction to fraud detection and prevention	11-12
6	Cybersecurity Regulations and Compliance in FinTech ● Exploring global and local (Ethiopian) regulations on cybersecurity in FinTech ● Understanding the role of compliance in cybersecurity	13-14
7	Cybersecurity Project ● Group project: Develop a cybersecurity and fraud prevention plan for a hypothetical FinTech company ● Presentation of group projects	15-16
Textbook and References:	1. Fintech Fundamentals: A Practical Guide to Understanding Financial Technology, 2023 by Mae Phelps 2. Financial Technology: Finance to create a better world for everyone, 2023. by Maxwell, Peter A. 3. A history of financial technology and regulation, 2022. By Oranburg, Seth. 4. Financial Technology: Case Studies in Fintech Innovation, 2020. by Niels Pedersen 5. Fintech in a Flash: Financial Technology made easy (3rd ed.), 2018. By Rubini,Agustin 6. Cybersecurity: The beginners guide: A comprehensive guide to getting started in cybersecurity, 2019. By Ozkaya, Erdal 7. Cybersecurity Essentials, 2018. By Brooks, Charles J.	
Particular resource req.:	None	
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.	

3.1.32 SWEN552 Computer Simulation and Modelling

Course Code:	SWEN552
Course Title:	Computer Simulation and Modelling
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Lab Hours:	2 hours per week
Course Schedule:	Academic Year V
	Semester II
Description:	The course will introduce the basic concepts of modeling and simulation that are increasingly being used by architects, planners, and engineers to shorten design cycles, innovate new products, and evaluate designs and simulate the impacts of alternative approaches. Students will use MATLAB to explore a range of programming and modeling concepts while acquiring those skills. They will then undertake a final project that analyzes one of a variety of scientific problems by designing a representative model, implementing the model, completing a verification and validation process of the model, reporting on the model in oral and written form, and changing the model to reflect corrections, improvements and enhancements.
Learning Outcomes:	Upon successful completion, students will be able to: ● Understand different methods for random number generation ● Have a clear understanding of the need for the development process to initiate the real problem. ● Have a clear understanding of principle and techniques of simulation methods informed by research direction. ● describe the components of continuous and discrete systems and simulate them ● model any system from different fields ● implement numerical algorithm to meet simple requirements, expressed in English ● identify suitable technique for specific simulation problem

Course Content

Unit	Topic	Week
1	Introduction to Modelling ● Nature of a model ● General Prerequisites ● Derivation of Models ● Analysis of Models ● Classification of Models	1-2
2	Introduction to Simulation ● General Remarks ● Assessment	
2	Modelling and Simulation Fundamentals ● Foundations ● Modelling and Simulation Process ● Verification and Validation ● Quality Assurance	3-4
3	Conceptual Modelling Framework ● Stochastic behaviour and modelling ● Data modelling ● Simulating Random Behaviour ● Exploring structural and behavioural requirements	5-7

Mid Semester Week		8
4	Statistical and mathematical concepts ● Elementary and Discrete topics ● Random number (Concepts, Algorithms for generation) ● Random number (Testing methods) ● Random Variable (Concepts, Types of distributions) ● Random Variable (Continue the types of distributions) ● Improving random variables (Variance reduction method, Layers method)	9-12
5	Practical Session ● Introduction to MatLab ● Game Theory, Group decision making, Traffie on highways, etc.	13-16

Textbook and References:	1. Modeling and Simulation in Python by Jason.M kinser (2024)
Particular resource req.:	Computer Lab, MatLab software
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.33 SWEN576 Management Information Systems

Course Code:	SWEN576
Course Title:	Management Information Systems
Prerequisites:	None
Credit Hours:	2 (4 ECTS)
Course Schedule:	Academic Year V Semester II
Description:	This course deals with the nature of information as an organizational resource; its significance in decision making and management; information requirements at different levels and functional areas of management; identification and source of information required; management of information system; development of information system; application of information and knowledge management. Common business applications related to management such as Financial Information Systems, Marketing Information Systems, Manufacturing and Production Information Systems, Human Resource Information Systems, Managerial Decision Support Systems will also be discussed.
Learning Outcomes:	The course enables students to: ● Explain the importance of MIS ● Describe the evolution & characteristics of the information age ● Understand and recognize the relationship between information and decision making, ● Know significance of information and information systems as basic resources from managerial perspective in decision-making. ● Appreciate the ability and skills to identify their information needs, source and to utilize the information efficiently and effectively.

Course Content

Unit	Topics	Week
1	Business management concepts ● Basic concepts and tools of strategic business management ● Developing competency in Business Management ● Factors for efficient strategic management ● Ethics in Business Management	1-2
2	Foundational Concepts in MIS ● Introduction ● Business and Management Functions ● The Information Needs and Sources of Managers ● A Framework for Information Systems ● Business Systems (e-business, e-commerce …) ● eBusiness value creation for management	3-4
3	IT Leadership and IS Strategic Planning ● IS Strategy and Effects of IT on Competition ● Re-engineering Work Processes for IT application ● Role of Internet and emerging technologies ● IT enabled services ● Seamless organizations ● Virtual corporations ● Web enabled computing as a strategic tool ● Outsourcing as a strategic alternative. ● International Information Systems	5-6
4	Securing Information Systems	7

	● Information Infrastructure ● Legal Issues and National Information Infrastructure. ● Factors contributing towards the IS security threats ● Technologies and Tools for protecting Information Resources	
Mid Semester Week		8
5	Common Business Applications of Information Technology ● Financial Information Systems ● Marketing Information Systems ● Manufacturing and Production Information System ● Human Resource Information Systems ● Managerial Decision Support Systems ● Transaction Processing System (TPS)	9-12
6	Knowledge Management (KM) ● Introduction to knowledge management ● Organizational Culture and Knowledge Management ● KM Tools and Technologies	13-16

Textbook and References:	1. Management information systems by issa, tomayess (2024) 2. Building life-cycle management, information system and technologies 1st edition by Ginzburg , alexander (2022)
Particular Resource Req.:	None
Teaching Strategy	Instructor delivers lectures, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.1.34 SWEN592 Software Engineering Capstone Project II

Course Code:	SWEN592
Course Title:	Software Engineering Capstone Project II
Prerequisites:	SWEN492
Credit Hours:	4 (7 ECTS)
Course Schedule:	Academic Year V
	Semester II
Description:	This course is a continuation from capstone project I of 1st semester.
Learning Outcomes:	The following are the learning outcomes of the capstone project: • Communication: In addition to written documentation of the project, students have the opportunity to develop their oral communication skills by way of providing presentations • Lifelong learning: Students will perform independent learning of new technologies and concepts • Modern Tools and Techniques: The completion of the project will enable students to select, and learn the necessary tools and techniques that are needed to complete the project.
Textbook and References:	Students will choose own reference materials based on the topic of their project.
Particular resource req.:	Computer Lab, students' choice of resources will be made available.
Teaching strategy:	Projects are carried out with continuous interaction between candidates and their designated supervisors.
Assessment:	The project is assessed through evaluation of the written report and the oral defense made by each candidate. An examination board set up for a project makes the assessment for each individual candidate.

3.1.35 MATH161 Discrete Mathematics

Course Code:	MATH161 Discrete Mathematics
Course Title:	Discrete Mathematics
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year I Semester I
Description:	This is an introductory course in discrete mathematics. The goal of this course is to introduce students to ideas and techniques from discrete mathematics that are widely used in science and engineering. The course teaches students techniques in how to think logically and mathematically and apply these techniques in solving problems. Students will learn Propositional logic and set theory, predicate Logic and quantification; the real and complex number systems; methods of proof (mathematical induction); relations and functions, sequences and series, arithmetic algorithms, computational complexity of algorithms and analytic geometry.
Learning Outcomes:	At the end of the course, students will be able to ● Understand and construct mathematical arguments ● Apply logical reasoning to solve a variety of problems ● Develop recursive algorithms based on mathematical induction ● Know basic properties of relations ● Understand basic concepts in formal languages and computability ● Apply knowledge about discrete mathematics in problem solving ● Use and interpret mathematically correct terminology and notation. ● Formulate a correct proof of a universally quantified statement. ● Propose a counter example to demonstrate that a statement is false. ● Know essential concepts in graph theory and related algorithms

Course Content

Unit	Topics	Week
1	The logic of compound statements ● Logical form and logical equivalence ● Conditional statements ● Validity and invalid arguments ● Application: Digital Logic Circuits ● Number Systems and Circuits for Addition,	1-2
2	The logic of quantified statements ● Predicates and Quantified Statements I ● Predicates and Quantified Statements II ● Statements with Multiple Quantifiers ● Arguments with Quantified Statements	3-4
3	Theory and concept of sets ● The language of sets ● Definitions and the element Method of proof ● Properties of sets ● Disproof's, Algebraic Proofs, and Boolean Algebras ● Boolean Algebra, Russell's Paradox, and the Halting Problem	5-6
4	Number theory & Methods of Proof, ● Direct proofs and counter examples ● Indirect Argument: -contradiction and contraposition ● Indirect Argument Two classical theorems ● Algorithms	7

Mid Semester Week		8
5	Relations and Functions ● Relations on Sets ● Equivalence Relations ● Partial Order Relations ● Functions Defined on General Sets	9-10
6	Exponential and Logarithmic Functions ▪ Exponents and radicals ● Exponential functions and their graphs ● Logarithmic functions and their graphs	11-12
7	Sequences, mathematical induction, and recursion ▪ Sequences - Summation Notation, Product Notation, ▪ Properties of Summations and Products, Factorial and "n Choose r" Notation, Sequences in Computer Programming, ▪ Application: Algorithm to Convert from Base 10 to Base 2 Using Repeated Division by 2	13-14
8	Trigonometry ▪ concept of functions ▪ combinations of functions ▪ Compositions of functions ▪ The trigonometric function ▪ Graph of the Trigonometric Functions ▪ Trigonometric inequalities and Equations ▪ Solving a Plane Triangle ▪ Solving any Triangle	15-16

Text book and References	The textbook for the course is Discrete Mathematics and its Applications, by Kenneth H. Rosen (McGraw-Hill, Inc., New York, 2018. (Or earlier editions) Textbook and References: Discrete Mathematics with Applications by Susanna S. Epp, 2010.
Particular Resource Req.:	None
Teaching Strategy:	Instructor delivers lectures, conducts tutorial sessions, prepares cases, reading assignments and problems for group discussion, gives consultation and advises students on assignment solutions, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: mid term exam and 50%: Final Examination.

3.1.36 MATH164 Linear Algebra

Course Code:	MATH164
Course Title:	Linear Algebra
Prerequisites:	MATH161
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year I
	Semester II
Description:	Linear algebra is the study of linear systems of equations, vector spaces, and linear transformations. Solving systems of linear equations is a basic tool of many mathematical procedures used for solving problems in science and engineering. In this course, students will concentrate on the mathematical theory and methods of linear algebra. Topics include systems of linear equations quadratic equations, functions, matrices and matrix algebra, inverse matrices; determinants and permutations; real n-dimensional vector spaces, abstract vector spaces and their axioms, linear transformations; inner products (dot products), orthogonality, cross products, and their geometric applications; subspaces, linear independence, bases for vector spaces, dimension, matrix rank; eigenvectors, eigenvalues, matrix diagonalization. Some applications of linear algebra will be discussed, such as economics, accounting, computer graphics, Kirchoff's laws, linear regression (least squares), Fourier series, or differential equations.
Learning Outcomes:	Upon completion of the course, students will ● Have good understanding of the concepts and methods of linear algebra, ● become competent in solving linear equations, performing matrix algebra, calculating determinants, and finding eigenvalues and eigenvectors. ● understand a matrix as a linear transformation relative to a basis of a vector space ● understand the concept of orthogonality of vectors and its use in projecting vectors into subspaces ● learn how to solve over constrained systems using the method of least squares ● connect linear algebra to other fields both within and without mathematics. ● develop abstract and critical reasoning by studying logical proofs and the axiomatic method as applied to linear algebra.

Course Content

Unit	Topics	Week
1	Complex numbers ● The set of complex numbers ● The complex plane ● De Moiré's theorem, powers and Roots	1
2	Vectors Space ● Definition of points in n-space ● Vectors and Geometry in two and three space dimensions ● Algebraic properties ● Dot Products and the norm of a vector ● Cross products and their geometric applications. ● Important inequalities	2-4

	● Vector Spaces, Subspaces and vector Space axioms ● Independence and orthogonal Vectors and Subspaces	
3	Matrices ● Definition of a matrix ● Algebra of matrices ● Types of matrices: square, identity, scalar, diagonal, triangular, symmetric, and skew symmetric matrices ● Elementary row and column operations ● Row reduced echelon form of a matrix ● Rank of a matrix using elementary row/column operations ● System of linear equations	5-7
Mid Semester Week		8
4	Determinants ● Definition of a determinant ● Properties of determents ● Adjoint and inverse of a matrix ● Cramer's rule for solving system of linear equations (homogenous and non-homogenous ● The rank of a matrix by sub determinants ● Determinant and volume ● Eigenvalues and eigenvectors of a matrix ● Diagonalization of a symmetric matrix	9-12
5	Linear Transformation ● Definition of linear transformations and examples ● The rank and nullity of a linear transformation and examples ● Algebra of linear transformations ● Matrix representation of a linear transformation ● Eigen values and eigenvectors of a linear transformation Eigen space of a linear transformation	13-16
Text book and References:	Text Book 1. Linear Algebra and Its Applications (5th Edition), 2015, by David C. Lay and Steven R. Lay 2. Introduction to Linear Algebra, Fifth Edition, 2016 by Gilbert Strang.	
Resource Req.:	None	
Teaching Strategy:	Instructor delivers lectures, conducts tutorial sessions, prepares cases, assignments and problems for group discussion, gives consultation and advises students on assignment solutions, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: mid term exam and 50%: Final Examination.	

3.1.37 MATH261 Calculus

Course Code:	MATH261
Course Title:	Calculus
Prerequisites:	MATH161
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year II
	Semester I
Description:	This course is designed to develop the topics of differential and integral calculus. Emphasis is placed on limits, continuity, derivatives and integrals of algebraic and transcendental functions of one variable. Rules of differentiation. Higher order derivatives. Chain rule. Related rates. Rolle's and the mean value theorem. Critical Points. Asymptotes. Curve sketching. Integrals. Fundamental Theorem. Techniques of integration. Definite integrals. Application to geometry and science. Indeterminate forms. L'Hospital's Rule. Improper integrals. Infinite series. Geometric series. Power series. Taylor series and binomial series.
Learning Outcomes:	Upon completion of the course, students will be able to ● Apply the definition of limit to evaluate limits by multiple methods and use it to derive the definition and rules for differentiation and integration. ● Use derivatives to analyze and graph algebraic and transcendental functions. ● Select and apply appropriate models and differentiation techniques to solve problems involving algebraic and transcendental functions; ● Apply the definition of indefinite integral to solve basic differential equations. ● Apply the definition of definite integral to evaluate basic integrals. ● Use the fundamental theorem of calculus to evaluate integrals involving algebraic and transcendental functions. ● select and use appropriate models and techniques for finding solutions to derivative-related problems.

Course Content

Unit	Topics	Week
1	Limits & Continuity ● Introduction to the limit concept ● Properties of limits ● Limits and infinity ● Continuity ● The intermediate value theorem (IVT) and its applications	1-3
2	Differentiations ● Definitions of derivative ● Tangent and normal lines ● Properties of derivative ● Derivative of different functions o polynomial, rational, trigonometric, exponential, logarithmic and hyperbolic functions ● The chain rule and parametric equations ● Higher order derivatives ● Implicit Differentiation	4-7

	● Extreme Values of Functions ● Rolle's Theorem and The Mean Value Theorem and their applications	
Mid Semester Week		8
3	Applications of Derivatives ● Rolle's Theorem and The Mean Value Theorem and their applications ● Monotonic Functions and the First and second derivative test ● Applications to extreme values and related rates ● Graph sketching and Tangent line approximation and the differentials ● Indeterminate Forms and L'Hôpital's Rule	9-10
4	Ant derivatives ● Indefinite integrals and their properties ● Partitions, upper sum, lower sum and ● Riemann sums ● The Definite Integral ● The fundamental Theorem of Calculus	11-13
5	Ant derivatives ● Indefinite integrals and their properties ● Partitions, upper sum, lower sum and ● Riemann sums ● The Definite Integral ● The fundamental Theorem of Calculus	14-16

Textbook and References:	1. Calculus 1 - Differentiation and Integration (Hamilton Education Guides Book 5), 2018 by Dan Hamilton 2. Advanced calculus 2nd edition by petrovic , john (2023)
Particular Resource Req.:	Graphic calculator
Teaching Strategy:	Instructor delivers lectures, conducts tutorial sessions, prepares cases, assignments and problems for group discussion, gives consultation and advises students on assignment solutions, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: mid term exam and 50%: Final Examination.

3.1.38 MATH266 Boolean Algebra

Course Code:	MATH266
Course Title:	Boolean Algebra
Prerequisites:	MATH164
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year II
	Semester II
Description:	This course covers the following topics: algebra of sets, basic Boolean functions, Boolean Expressions and Truth Tables, digital logic gates, minterm and maxterm expansions, the basic theorems of Boolean algebra, simplifying Boolean function with karnaugh maps. Relay circuits and control problem, circuits for arithmetic competition, probability in finite sample space.
Learning Outcomes:	On completing this course, students will be able to ● prove a number of useful basic theorems from given Boolean axioms; ● simplify and complement Boolean expressions; ● define the fundamental logic operations AND, OR, Invert; ● relate Boolean expressions to truth tables and logic diagrams. ● Use truth tables and laws of identity, distributive, commutative, and domination. ● Simplify and prove Boolean expressions ● Compute sum of products and product of sum expansions. ● Convert Boolean expressions to logic gates and vice-versa.

Course Content

Unit	Topics	Week
1	Algebra of Sets ● Introduction ● Elements and Sets ● Combination of sets ● Venn Diagram ● Fundamental Laws ● Expanding, Factoring and Simplifying ● Properties of Sets inclusion ● Conditional Equations ● Solution of Equations ● Number of Elements in a set	1-3
2	Boolean Algebra ● Preliminary Definitions ● Definitions and properties of Boolean Algebra ● Disjunctive normal form ● Conjunctive normal form ● Representation of a Boolean Algebra	4-5
3	Symbolic Logic and Algebra of Propositions ● Propositions and definitions of symbols ● Truth table ● Object logic and syntax logic ● Material implication ● Truth sets for propositions ● Quantifiers ● Valid arguments	6-7

	● Indirect truth ● Functionally complete set of operations	
Midsemester Week		8
4	Switching Algebra ● Definition of algebraic symbols ● Simplification f circuits ● Non-series parallel circuits ● Design of circuits from given properties ● Design of n terminal circuit ● Symmetric functions and their circuits	9-10
5	Relay circuits and control problem ● Basic relay control path ● N terminal circuits and the use of transfer contacts ● Operate and hold paths ● Sequential circuits and sequence diagram ● Design of sequential relay circuits from given conditions	11-12
6	Circuits for Arithmetic competition ● Binary number system ● Logical circuit elements ● Addition of Binary numbers ● Subtraction of Binary numbers ● Accumulation ● Binary multiplication	13-14
7	Probability in Finite sample space ● Events, sample space, probability ● Conditional probability ● Some aids to counting ● Bernoulli trials, binomial distribution	15-16
Textbook:	1. Boolean algebera by ufuoma, okoh (2020)	
Particular Resource Req.:	Graphic calculator	
Teaching Strategy:	Instructor delivers lectures, conducts tutorial sessions, prepares cases, assignments and problems for group discussion, gives consultation and advises students on assignment solutions, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: mid term exam and 50%: Final Examination.	

3.1.39 MATH361 Statistical Methods

Course Code:	MATH361
Course Title:	Statistical Methods
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year III
	Semester I
Description:	This course offers a comprehensive introduction to the fundamental principles of statistical methods. Students will learn to collect, analyze, interpret, and present data effectively. The course will cover descriptive statistics, probability, sampling distributions, hypothesis testing, confidence intervals, correlation, and regression analysis. Practical applications and real-world examples will be used to illustrate concepts and techniques..
Learning Outcomes:	By the end of this course, students will be able to: ● Understand and apply basic statistical concepts and methods. ● Collect, summarize, and interpret data. ● Perform and interpret various statistical tests. ● Use statistical software to analyze data. ● Communicate statistical findings effectively.

Course Content

Unit	Topics	Week
1	Statistics and Scientific Methods ● Why study statistics ● Application of statistics ● Types of Statistics	1
2	Data Collection and Presentation ● Methods of Data Collection ● Data Organization and Presentation ○ Tabular and Graphical Representations	2-3
3	Measure of Central Tendency ● Mean, Median Model ● Properties and Applications	4-5
4	Measures of Dispersion ● Range, Variance, Standard Deviation ● Interpretation and use	6-7
Mid Semester Week		8
5	Measures of Dispersion ● Range, Variance, Standard Deviation ● Interpretation and use	9-10
6	Probability and probability distribution ● Probability rules ● Probability distribution for discrete random variables ● Probability distribution for continuous variables	11-12
7	Foundations for inference ● Estimation of Variables ● Confidence intervals ● Hypothesis testing	12-13
8	Correlation and Regression ● Correlation Analysis ● Simple Linear Regression	14-16

	● Multiple Regression
Textbook and reference	1. An Introduction to Statistical Methods and Data Analysis, 2021 by R. Lyman Ott and Micheal T. Longnecker 2. Introduction to Probability (Chapman & Hall/CRC Texts in Statistical Science),2019, by Joseph K. Blitzstein and Jessica Hwang Simon
Particular Resource Req.:	Statistical packages, eg. SPSS, Excel
Teaching Strategy:	Instructor delivers lectures, conducts tutorial and lab sessions, prepares cases, assignments and problems for group discussion, gives consultation and advises students on assignment solutions, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: midterm exam and 50%: Final Examination.

3.2 Elective Courses

3.2.1 SWEN437 Computer Graphics

Course Code:	SWEN437
Course Title:	Computer Graphics
Prerequisites:	
Credit Hours:	3 (5 ECTS)
Lab Hours	2 hours per week
Course Schedule:	Academic Year IV Semester I
Description:	The aim of this course is to provide a unified introduction to computer graphics and computer vision It introduces both the hardware and software utilized in computer graphics imaging or digital visual arts. The emphasis is on creating a working graphics system from the ground up, but modern models and applications are also discussed and utilized. The fundamentals of display hardware and applications, interactive techniques and color models, study of 3D viewing pipeline, drawing images in OpenGL, 3D polygon rendering and transformations are discussed shrewdly.
Learning Outcomes:	Upon successful completion of this course, students should be able to: ● create interactive graphics applications. ● Have a knowledge and understanding of techniques for representing 3D geometrical objects ● Understand the fundamental concepts of rendering with openGL ● Model 3D objects using polygons. ● Understand non-polygon representation of objects and realize the difference between the above. ● Have a knowledge and understanding of the fundamental principles of application modeling.

Course Content

Unit	Topic	Week
1	Introduction to interactive computer graphics ● Brief History of Computer Graphics ● 3D Graphics Techniques and Terminology ● Common Uses of Computer Graphics ● Examples of application areas	1
2	Graphics hardware ● Raster display systems ● Introduction to the 3D graphics pipeline	2
3	Introduction to the rendering process with OpenGL ● The role of OpenGL in the reference model Coordinate systems ● Viewing using a synthetic camera ● Output primitives and attributes.	3-4
4	Geometry and Line Generation ● Point and Lines, Bresenham's ● algorithm Generating Circles ● Plotting General Curves ● Line Thickness ● Line Style ● Polygons	5-6

	● Filling ● Text and Characters	
5	Geometrical Transformations ● 3D transformation ● Matrix representation ● Homogeneous coordinates ● Combination of transformations	7
Mid Semester Week		8
6	State Management and Drawing Geometric Object ● Basic State management ● Displaying Points Lines and Polygons ● Normal Vector ● Vertex Arrays	9
7	Representing 3D objects ● Modeling using polygons ● Techniques for creating representational polygonal meshes ● Non-polygonal representations	10
8	Colors and Images ● Colour in Computer graphics RGB; CIE ● Image formats and their applications: GIF, JPG, PNG.	11-12
9	Viewing A local illumination model ● Using the camera model for viewing 3D scenes ● Perspective and other types of projection ● Viewing Types of light source ● Reflectance models: diffuse (Lambert) and specular (Phong) Gouraud and Phong interpolation ● Lighting and shading in OpenGL Textures	13-14
10	Application modeling ● Distinction between Modeling and graphics ● Immediate mode versus retained mode Model ● Storage Strategies ● OpenGL display lists: traversal, Instancing	15-16

Textbook and References:	1. Developing graphics frameworks with java and OpenGL by stemkoski , lee (2022) 2. Fundamnetals of computer graphics 5th edition by marschner steve (2023)
Particular resource req.:	OpenGL should be installed on all laboratory Windows machines and Linux. Assignments may be done on either Windows (using Visual Studio .NET or Net beans) or Linux (using gcc/g++).OpenGL does not have calls to deal with user interaction, like mouse clicks or opening windows.
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.2.2 SWEN439 Game Development

Course Code:	SWEN439
Course Title:	Game Development
Prerequisites:	
Credit Hours:	3 (5 ECTS)
Lab Hours:	2 hours per week
Course Schedule:	Academic Year IV
	Semester I
Description:	This course introduces you to Games development, emphasising a mix of creative content design, development,and technical specialisation. You will gain an understanding of the Games industry from its conception through to current trends. Initially, you will study the lifecycle of games development, focusing on story design, character design, game mechanics, and level design, as well as content development including textures and interface, 3D modelling, game development, and programming. You will learn event driven programming through triggers and updates in a games development environment. Via lectures and hands-on projects, the course explores principles of 2D and 3D graphics, animation, sound, and collision detection using various frameworks, as well as programming languages. By class's end, students will have programmed several of their own games and gained a thorough understanding of the basics of game design and development.
Learning Outcomes:	Upon successful completion of this course, students should be able to: ● Describe the games industry, technologies and cultures; ● Discuss games design and development methodologies; ● Identify and explain the appropriate and correct syntax and programming constructs for different game development requirements. ● Select and apply appropriate games design and development approaches to align with industry needs; ● Design and develop a range of art and programming assets, implementing aesthetics and logic into a game project; ● Analyse, design, implement and test game concepts using a games engine and programming constructs; ● Utilize appropriate software packages to design, build and program game prototypes

Course Content

Unit	Topic	Week
1	Introduction to Game Design ● Definitions, characteristics and mechanics of games. ● History and Generations of Games. ● Uses and applications of games related skills and practices ● Game Genre Overview	1-2
2	Game Design Principles ● Layers of Game Design ● Game character design ● Game story development ● Gameplay Development ● Creating successful game	3-5
3	Game Assets	6-7

	● Gaming asset modelling ● Gaming assets services ● 2D and 3D asset development	
Mid Semester Week		8
4	Game Development Tools ● Tools and applications used for game development	9-11
5	Game Balance ● Game balance methodologies ● Balancing game economies ● Dynamic game balancing	12-13
6	Game Design Project	14-16

Textbook and References:	1. Game development 2042 by fields , tim (2022) 2. Introduction to game design ,prototyping and development 3rd edition by Gibson bond, Jeremy (2022)
Particular resource req.:	Computer lab, Python programming language
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.2.3 SWEN466 Knowledge Discovery and Data Mining

Course Code:	SWEN466
Course Title:	Knowledge Discovery and Data Mining
Prerequisites:	SWEN364
Credit Hours:	3 (5 ECTS)
Lab hours	2 hours per week
Course Schedule:	Academic Year IV Semester II
Description:	This course discusses basics of the knowledge discovery process, data mining, and provides a basic introduction to data science. It also presents current research in Knowledge Discovery in Databases (KDD) dealing with data integration, mining, and interpretation of patterns in large collections of data. Topics include data warehousing and data pre-processing techniques; data mining techniques for classification, regression, clustering, deviation detection, and association analysis; and evaluation of patterns mined from data. Industrial and scientific applications are discussed.
Learning Outcomes:	At the end of the course, students will be able to: ● Define, describe, and clearly state the objectives of Knowledge Discovery and Data Mining. ● Understand how to implement common data mining techniques to extract patterns, trends, and other useful information from databases. ● Identify relevant data and corresponding databases and data warehouses. ● Mine and discover models, patterns, dependencies that will enable predictions, and make intelligent business and operation decisions, ● Present and document results.

Course Content

Unit	Topics	Week
1	Introduction ● Meaning of Data Mining ● Essence of Data Mining ● Relationship between Data Mining, Data ● Warehousing and On-line Analytical Processing ● Issues in Data Mining ● The KDD/DM Process Model; Prediction vs. Description modeling	1-2
2	Data warehousing and OLAP Technology for data mining ● OLAP technology, attribute-oriented induction ● What is a data warehouse? ● A multidimensional data model ● data cube computation ● Data warehouse architecture ● Data warehouse implementation ● From data warehouse to data mining	3-4
3	Data preprocessing ● Why preprocess data? ● Major Tasks in Data Preprocessing o Data Exploration o Data understanding o Data cleaning and reduction	5-7

	○ Data Integration and Transformation ○ Discretization and concept hierarchy generation	
Mid Semester Week		8
4	Classification and prediction ● Meaning of Classification and prediction ● Issues regarding classification and prediction ● Classification by decision tree induction ● Bayesian classification ● Classification by back propagation ● Other classification methods ● Prediction ● Classifier accuracy	9-11
5	Cluster analysis ● What is cluster analysis? ● Types of data in cluster analysis ● Categorization of major clustering methods ● Partitioning methods ● Hierarchical methods ● Density based methods & Outlier analysis	12-14
6	Mining association rules in large databases ● Overview of Pattern Discovery ● Pattern finding and association rules discovery techniques	15-16

Textbook and References:	1. Data Science for Business: Predictive Modeling, Data Mining, Data Analytics, Data Warehousing, Data Visualization, Regression Analysis, Database Querying, and Machine Learning for Beginners, 2018, by Herbert Jones 2. Artificial intelligence and data mining approaches in security frameworks by Bhargava, neeraj (2021).
Particular Resource Req.:	WEKA Data Mining Tool, Python Programming Environment, R Programming language, Other appropriate data mining and data warehousing tools shall also be selected by the instructor
Teaching Strategy	Instructor delivers lectures, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.2.4 SWEN468 Software Agent

Course Code:	SWEN468
Course Title:	Software Agent
Prerequisites:	
Credit Hours:	3 (5 ECTS)
Lab hours	2 hours per week
Course Schedule:	Academic Year IV
	Semester II
Description:	The course introduces agent systems and software agents. It focuses on agent system architecture and infrastructure from a software engineering viewpoint, including requirements for agent-based systems; modeling and design of agent-based systems and the development process for agent-based systems. Topics such as agent architecture, communication, knowledge sharing, computing, and uncertainty management are discussed. Studying society of agents and models of agency follows. Finally, a perspective on a methodology for agent-oriented software engineering and standards are presented.
Learning Outcomes:	At the end of this course the students will be able to: ● Understand what the myths and realities of the agent-based systems are ● to develop an agent-based system for a particular task ● get the exposure to evolve from object-oriented development to agent-based systems ● understand ways of incorporating and sharing knowledge among software agents?

Course Content

Unit	Topic	Week
1	Overview of agent-based software engineering.	1-2
2	Methodologies for agent-based modeling, analysis and design.	2-4
3	Agent communication and knowledge sharing. ● Overview of Agent Programming ● Agent Communication language ● Agent Based Framework of Interoperability	5-7
Mid Semester Week		8
4	Agent-based System Architecture and Organization. ● Agent for Information Gathering ● Open Agent Architecture ● Communicative Action for Artificial Agent	9-11
5	FIPA: Foundation for Intelligent Physical Agents.	12-14
6	Mobile Agents ● Mobile Agent Paradigm ● Mobile Agent Concepts ● Mobile Agent Technology	15-16

Textbook and References:	1. Software agent (2nd ed.), 2022, by Gerardus Blokdyk | Jan 16, 2022 2. Software Agents, 2017 by Bradshaw, Jeffrey M. (ed.) The MIT Press. 3. Software Agents Based Web Data Mining: An Intelligent Approach, 2023 by Kundu, Shakti Instructor selects articles to be read by students.

Particular resource req.:	Computer Lab
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.2.5 ITSY481 Cloud Computing and Data Centre Management

Prerequisites:	IT481
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year V
	Semester II
Description:	This course provides basic concepts of cloud computing and data centre management: a hands-on comprehensive study of Cloud computing and capabilities across the various Cloud service models including Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), and Business Process as a Service (BPaaS), awareness of data center requirements, design and management technologies and methodologies. The course also gives insight on data center operational characteristics and requirements as well as industry best practice principles for achieving effective operational management.
Learning Outcomes:	Upon completion of the course, students will have ● Knowledge of the main concepts, key technologies, strengths, and limitations of cloud computing ● Exploit the benefits of the different cloud service models: SaaS, PaaS, and IaaS ● explain the core issues of cloud computing such as security, privacy, and interoperability. ● Have a broad overview of data centre technology and the interactions and interdependencies of data centre components ● knowledge in data centre infrastructure, operations and best practices.

Course Content

Unit	Topic	Week
1	Cloud Computing Overview ● Cloud Computing definition and characteristics ● Cloud Computing and SOA ● Enterprise Cloud drivers and adoption trends ● Cloud service models/types (public, private, hybrid, and community clouds) ● Cloud deployment models ● Cloud reference architectures	1-3
2	Cloud Computing Services ● Infrastructure as a service (Iaas) ● Platform as a service (PaaS) ● Software as a Service (SaaS) ● Business Process as a Service (BPaaS)	4-7
Mid Semester Week		8
3	Cloud Security ● Cloud security challenges ● Cloud security approaches: ● Design of secured cloud architecture	9-10
4	Planning Cloud transformations ● suitability assessment, ● financial assessment and platform selection, ● roadmap definition	11-13
5	Data Centre Management ● Overview of Data Centres ● Components of Data Centre	14-16

	o Network infrastructure. ... o Storage infrastructure. ... o Computing resources. ... o Network security appliances. ..	
Textbook and References	1. Cloud Computing: An Introduction, 2017 by by R. Chopra 2. Handbook of Data Center Management: Second Edition (CRC Press Revivals), 2017, Wayne C. Bradley 3. Cloud Security: Introduction to cloud security and data protection, 2018, by Nate Jenner 4. Data Center Infrastructure & Organization, 2016, by George Haynes 5. Cloud industry publications, online Textbook, and research papers on various topics connected to the various sessions	
Particular Resource Req.:	Computer lab, visit to data centre and hands on experiment	
Teaching Strategy	Instructor delivers lectures, conducts lab sessions, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination	

3.2.6 SWEN554 Ethical Computing

Course Code:	SWEN427
Course Title:	Ethical Computing
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year V
	Semester II
Description:	This course aims to introduce student to moral principles or values that define or direct the right choice. Topics covered include defining of ethics, personal vs. professional ethics, Code of ethics, professional practices; Acting ethically, ethical obligations to the public, case studies on ethical implications of online harassment for software engineers; Case studies on privacy, ethical implications of blindly following customers' requirements.
Learning Outcomes:	At the end of the course, students will be able to: ● understand the need for both personal and professional ethics. ● analyze the ethical implications of software engineering practices that can incur harm. ● understand how ethical practice involves not just avoiding harm, but doing good. ● apply five ethically constructive habits of mind and action. ● get awareness on codes related to ethics and can apply them to their practice. ● apply ethical principles to controversies such as online harassment and privacy.

Course Content

Unit	Topic	Week
1	Introduction to cybernethics ● Definition of key terms ● Cyberethics evolution ● Cyberethics methodology	1
2	Ethical concepts and ethical theories ● Ethics and morality ● Ethical theories	2
3	Professional ethics, code of conduct, and moral responsibility ● Professional ethics ● ITSY professionals' special moral responsibilities ● Moral responsibility, legal liability and accountability	3-4
4	Privacy and cyber space ● Privacy in the digital age ● Personal privacy ● Gathering personal data: surveillance, recording and tracking techniques ● Internet cookies ● RFID technology ● Analysing personal data ● Protecting personal privacy ● The right to "Be Forgotten"	5-7
Mid Semester Week		8
5	Security in cyber space ● Data security ● System security	9-10

	● Network security ● Hacking and hacker ethic ● Cyber terrorism ● Hacktivism	
6	Cyber crimes ● Cyber crimes and cyber criminals ● Hacking, cracking and counter hacking ● Combatting cyber crimes ● Biometric technologies	11-13
7	Intellectual property disputes ● What is intellectual property ● Copyright law and digital media ● Patents, trademarks and trade secrets ● The opensource movement	14-16

Textbook and References:	1. Ethics and Technology: Controversies, Questions, and Strategies for Ethical Computing, 2023, by Herman T. Tavani There will also be supplemental readings beyond the References Textbooks, such as articles or web pages, which will be assigned by the instructor throughout the semester.
Particular resource req.:	None
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.3. Support Courses

3.3.1 SPRT111 College English I

Course Code:	SPRT111
Course Title:	College English I
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year I
	Semester I
Description:	This course is intended to develop and improve students' language competence. It is also aimed at developing students' communicative abilities in English which will help students to develop their communicative skills and overall language competence in English. Generally, this course will cover the specific language aspects described below. Developing basic functions of English language skills: **reading** (scanning, skimming, reading for details, summarizing, understanding the structure of a text); **listening** (listening for the gist, listening for details, recognizing discourse markers, noticing the structure of a lecture, understanding speaker intentions, recognizing signposting, attending and following skills); **writing** (summarizing a text, writing descriptive texts); **speaking** (introducing oneself and others, interviewing, discussions, stating and supporting propositions, stating one's opinions, organizing and taking part in a debate, making a persuasive speech, questioning); **vocabulary** (working out meanings from context, synonyms, antonyms, collocations, definitions); **grammar** (relative clauses, modals, voice, conditionals, tense, reported speech).
Learning Outcomes:	Upon completing the course, students will be able to: ● Express their ideas in various communicative contexts (in group/ pair discussion, public speaking settings etc.) ● Use various vocabulary learning strategies and techniques ● Write and present reports ● Read various materials and make their own notes ● Identify the structure of oral and written discourses ● Attend their academic work at ease and with clarity.

Course Content

Unit	Topic	week
1	Introductions: Course; Instructor, students, working procedures ● Searching about people and events; how to learn about vocabularies; Punctuation Marks: Capitals, Apostrophes, Semicolons, Colons, commas, quotation marks, full-stop and question marks; Introduction to College English; set of demands; Discussion: First impression of College study and the demands; Writing a short description about self-selected topic; Redraft based using comments. Study Skills ● Attending lectures, Taking short notes, Improving notes through group interaction; Building vocabulary; language and meaning, Negation and expansion; Articles on study skills; Different reading skills: Scanning, skimming, reading for details, understanding structure of texts; Telling the gist of a lecture; Improving Writing Skills:	1-3

	basic types of writing –Expository, Narrative, Descriptive and Argumentative types	
2	Health and Fitness ● Article about current situation of COVID 19, Markers of addition and relating; Using components of a word as clues to its meaning; Collocation, context and relationships to topics to learn new words; Relative clauses; defining and non-defining relative clauses; Critical Reading; Public Speaking: Preparing and making short talk about the thematic issue; Producing a fact sheet; writing a transcript for a radio broadcast; Argumentative writing.	4-5
3	Current Development on Information Technology ● Active Listening: Fighting Challenges like daydreaming, detouring, private planning; Dictionaries for references, word formation and parts of speech; Speech acts and grammar; Critically analyzing and appreciating poems / short stories. Reading for details; Adding variety to your speech; brainstorming; Summarizing a talk or text, commenting on academic articles; Writing in direct/ indirect forms	6-7
Mid Semester Week		8
4	Cultural Values ● Identifying structure of lectures; Follow markers to get main ideas; Learning meaning of words from their origins: Latin or Greek; Using active and passive voices, degrees of frequency; Time clauses; Reading for main Ideas; reading for details; understanding references; Taking part in debate; Summarizing key ideas from a text. writing descriptive Essay	9-12
5	Current transformations in the World ● Identifying New Developments; reason out for observed changes; Completing vocabulary network; Conditionals: 1,2 and 3; Ways of expressing cause and effect; Looking at the purposes of introduction and conclusions; Speaking to convince listeners; public speaking; Write Argumentative essay	13-16
Textbook and References:	1. McGraw-Hill Handbook of English Grammar and Usage, 2nd Edition, 2012, by Mark Lester and Larry Beason 2. Basic English Grammar Workbook, Feb 20, 2014, by Betty S. Azar and Stacy A. Hagen **3.** Just the Basics of English Grammar: A workbook for the most common writing problems, 2014, by Sheldon Lawrence Ph.D.	
Teaching Strategy:	Articles for listening exercise of all units need to be selected and organized. Each Unit is finalized with student Reflection and self-assessment which is checked by the instructor. Instructor delivers lectures, prepares discussion sessions with students, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.	

3.3.2 SPRT112 College English II

Course Code:	SPRT112
Course Title:	College English II
Prerequisites:	SPRT111
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year I Semester II
Description:	College English II is a continuation of College English I, and it mainly aims to provide first year College students proficiency with reading, speaking and writing skills that will be of use for the academic work expected from each student in their higher education career and thereafter. It also aims to help students learn vocabularies that are assumed unfamiliar to them. In the grammar part, with the intention of providing explanations, brief notes are given in each unit. The module consists of five units with three supplementary reading at the end of the Module. The supplementary readings are included to support ideas included in the reading passages in units 1-3. Students are advised to read the references or notes put in the box to further learn the grammar topics included in the Module.
Learning Outcomes:	Upon completing the course, students will be able to: ● Identify different components of 'life skills' so that they can actively apply them in life; ● Understand how scientific investigation can be carried out; ● Express their ideas in various communicative contexts (in group/ pair discussion, public speaking settings etc.) ● Use various vocabulary learning strategies and techniques ● Use the future tense forms, in their speech and writing, when appropriate. ● Become aware of the environmental problems and how they can be resolved; ● Determine to participate in environmental protection activities; ● Develop their speaking, listening, reading and writing abilities; ● Use modal verbs, direct and indirect speech in academic discussions and academic writing. ● Be aware of the importance of indigenous knowledge and cultural heritage;

Course Content		
Unit	Topic	Week
1	**Life Skills** ● Reading Passages preceded and followed by students exercises to be done outside class; discussion of the correct answers in class, making corrections for mistakes done. ● There are notes on types of conclusions in easy writing - the embedded, the retrospective, the reflective and the projective. Examples are given for each type. Students practice writing conclusions of each type. ● Active and passive voices in different tense are exercised followed by writing paragraph, vocabularies in the garment production process and speaking exercise to improve interpersonal skills	1-3
2	**Speculations about the Future of Science** ● Student Activities: Reading passage on Grassroots attack in bilharzias preceded by pre-reading questions and followed by comprehension and reflective questions to be done by students: classroom discussion on the answers for the questions. Preparing and making short talk about the thematic issue	4-5

	● There are new vocabularies used in the passage that the readers are expected to comprehend from the context. ● A tabled note on the different forms and functions of the future tense in English is given with work-on. ● Speaking and writing activities conclude the unit student exercises.	
3	**Environmental Protection** ● Pre reading questions followed by a passage on environmental challenges is offered. The grammar part deals with modal verbs: can, could, may, might, must, shall, should, ought to, will, and would. Notes on Modal verbs and their functions are given followed by exercises. There are speaking and writing exercises on debatable environment issues where students are expected to argue raising causes and solutions. ● The grammar part deals with identifying quoted and reported speech; changing from direct to indirect speech.	6-7
Mid Semester Week		8
4	**Indigenous Knowledge** ● There is a passage entitled "A Local Pathway to Global Development" written by Benjamin Mkapa, where selected vocabulary are highlighted for study and comprehension exercises are placed amid the passage for students' reflective activities. ● The grammar exercise is on reported speech and direct and indirect speech followed by speaking exercises through group discussion and writing argumentative paragraphs the thematic issues	9 – 12
6	**Cultural Heritage** ● The last unit for the course deals with cultural heritage where students will be guided to identify man-made and natural heritages; be aware of the importance of cultural heritages in national development; develop their speaking, listening, reading and writing abilities; and understand and use relative clauses in their oral and written discourses. ● Notes are given for defining and non-defining relative closes followed by speaking and writing exercises. ● Additionally, there are supplementary reading materials attached to the module for units 3 to 5 to enrich students understanding about themes of the units.	13 – 16

Textbook and References:	**1.** Azar, B. S. (2003). Fundamentals of English grammar. Longman. Eggenschwiler, **2.** J.,& Biggs, E.D. (2001). Writing: Grammar, Usage, and Style. New York. Hungry Minds. Inc Lucy, J. A., & Lucy, L. A. (Eds.). (1993). **3.** Reflexive Language: Reported Speech and Meta pragmatics. Cambridge University Press. 4. Murphy, R. (2012). English Grammar in Use. Ernst Klett Sprachen. Naylor, H., & Murphy, R. (2007). Essential Grammar in Use. Supplementary Exercises. With Answers. Ernst Klett Sprachen
Particular Resource Req.:	None
Teaching Strategy:	Each Unit begins with statements of learning outcomes followed by probing questions to activate student critical thinking. Then reading passages on thematic issues of the unit, vocabularies, grammar and writing exercises follow with intermittent reflective exercises. Each unit is finalized with student Reflection and self assessment which is checked by the instructor.

	Instructor delivers lectures, prepares discussion sessions with students, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.3.3 SPRT115 Geography of Ethiopia and the Horn

Course Code:	SPRT115
Course Title:	Geography of Ethiopia and the Horn
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year I
	Semester I
Description:	This course intends to familiarize students with the basic geographic concepts particularly in relation to Ethiopia and the Horn of Africa. It is also meant to provide students a sense of place and time (geographic literacy) that are pivotal in producing knowledgeable and competent citizens who are able to comprehend and analyze spatial problems and contribute to their solutions. the course provides an opportunity for the reader to understand the implications of the location, shape and size of Ethiopia, as well as the country's physical and human resources diversity and abundance on its socioeconomic development. Main focuses of the course are: Shape and size of Ethiopia; basic skills of reading maps; physical background and natural resource endowment of Ethiopia and the Horn which includes its geology and mineral resources, topography, climate, drainage and water resources, soil, fauna and flora; demographic characteristics of the country and its implications on economic development; treatment of the various economic activities of Ethiopia and the Horn which include agriculture, manufacturing and the service sectors.
Learning Outcomes:	Upon completing the course, students will ● Have basic familiarity on the location shape and size of Ethiopia and the Horn ● Have basic familiarity on the Topography and Geology of Ethiopia and the Horn ● Be able to identify the rocks and mineral resources of Ethiopia ● Have understanding of Topography of Ethiopia and the horn ● Be familiar with Drainage systems and water resources of Ethiopia and the horn ● Be familiar with the Natural vegetation and wildlife resources of Ethiopia ● Have understanding of the climate of Ethiopia and the Horn ● Have Population of Ethiopia and the horn ● Have basic skills of demographic measurements

Course Content

Unit	Topics	Week
1	Introduction ● Geography: Definition, Scope and Themes ● Location, Shape and Size of Ethiopia and the Horn ● Basic Skills of Map Reading	1-2
2	The Geology of Ethiopia and the Horn ● The Geologic Processes: Endogenic and Exogenic Forces ● The Geological Time Scale and Age Dating Techniques ● Geological Processes and the Resulting Landforms of Ethiopia and the Horn ● Rock and Mineral Resources of Ethiopia	3-4
3	The Topography of Ethiopia And the Horn	5-6

	● The Physiographic Divisions of Ethiopia ● The Impacts of Relief on Biophysical and Socioeconomic Conditions	
4	Drainage Systems and Water Resource of Ethiopia and The Horn ● Major Drainage System of Ethiopia ● Water Resources: Rivers, Lakes and sub-surface water ● Water Resources potentials and Development in Ethiopia	7
Mid Semester Week		8
5	The Climate of Ethiopia and The Horn ● Elements and Controls of Weather and Climate ● Spatiotemporal Patterns and Distribution of Temperature and Rainfall in Ethiopia ● Agro-ecological Zones of Ethiopia ● Climate Change/Global Warming: Causes, Consequences and Response Mechanisms	9-10
6	Soils, Natural Vegetation and Wildlife Resources Of Ethiopia And The Horn ● Ethiopian Soils: Types, Degradation and Conservation ● Natural Vegetation of Ethiopia ● Wild Life/wild animals in Ethiopia	11
7	Population of Ethiopia And the Horn ● Population Data: Uses and Sources ● Population Dynamics: Fertility, Mortality and Migration ● Population Distribution in Ethiopia ● Socio-cultural Aspects of Ethiopian Population: Education, Health and Languages ● Settlement Types and Patterns	12-13
8	Economic Activities in Ethiopia ● Mining Activity in Ethiopia ● Forestry ● Fishery ● Agriculture in Ethiopia ● Manufacturing Industry in Ethiopia ● The Service Sector in Ethiopia	14-16
Textbook and References:	1. Geography of Ethiopia and the Horn. Compiled by Dr. Teferri Mekonnen et al. September 2019 (Handout/reference prepared for students of higher learning in Ethiopia)	
Teaching Strategy:	Lectures, reading assignments, discussions with students, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment,20%: Project term paper and 50%: Final Examination,	

3.3.4 SPRT117 Introduction to Logic and Critical Thinking

Course Code:	SPRT117
Course Title:	Introduction to Logic and Critical Thinking
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year I Semester I
Description:	This course is designed to acquaint students with the terms, problems, methods, and theories of several different areas within philosophy. It will introduce students to the major topics of philosophy, explores such fundamental issues as metaphysics, epistemology, political philosophy, ethics, and the philosophy of religion. The course aims to improve students' ability to think critically, develop ideas and express these ideas clearly and persuasively in writing. The course is designed to help students develop the abilities and skills of critical thinking and to construct reliable and logically defendable arguments of their own and rationally evaluate the arguments of others.
Learning Outcomes:	Upon successful completion of this course, students will be able to: ● Understand the basic essence and areas of philosophy, and the necessity of learning it; ● Recognize the components and types of arguments; ● Develop the skill to construct and evaluate arguments; ● Understand the relationship between logic and language; ● Recognize the forms of meanings of words and terms; ● Comprehend the types, purposes and techniques of definitions; ● Understand the concept, principles, and criteria of critical thinking; ● Cultivate the habits of critical thinking and develop sensitivity to clear and accurate usage of language; ● Recognize the various forms of formal and informal fallacies; and ● Understand the components, attributes and representations of categorical propositions.

Course Content

Unit	Topics	Week
1	Introducing Philosophy ● Meaning and Nature of Philosophy ● Basic Features of Philosophy ● Core Fields of Philosophy ● Metaphysics and Epistemology ● Axiology and Logic ● Importance of Learning Philosophy	1-3
2	Basic Concepts of Logic ● Basic Concepts of Logic: Arguments, Premises and Conclusions ● Techniques of Recognizing Arguments ● Types of Arguments: Deduction and Induction ● Evaluating Arguments	4-5
3	Logic and Language ● Lesson 1: Philosophy of Language: An overview ● Logic and Meaning ● Logic and Definition ○ Meaning, Types, and Purposes of Definitions	6-7

	○ The Meaning of Definition ○ The Types and Purposes of Definitions ○ Techniques of Definition ○ Criteria for Lexical Definitions	
Mid Semester Week		8
4	Basic Concepts of Critical Thinking ● Meaning of Critical Thinking ● Standards of Critical Thinking ● Codes of Intellectual Conduct for Effective Discussion ● Characteristics of Critical Thinking ● Barriers to Critical Thinking ● Benefits of Critical Thinking	9-10
5	Informal Fallacies ● Fallacy in General ● Informal fallacies ● Fallacies of Relevance ● Fallacies of Weak Induction ● Fallacies of Presumption ● Fallacies of Ambiguity and Grammatical Analogy	11-13
6	Categorical Propositions ● General Introduction ● Attributes of Categorical Propositions: Quality, Quantity, and Distribution ● Venn Diagrams and the Modern Square of Opposition ● Evaluating Immediate Inferences: Using Venn Diagrams and Square of Oppositions ● Logical Operations: Conversion, Obversion, and Contraposition	14-16
Textbook and References:	1. Introduction to Philosophy: Classical and Contemporary Readings, 2016, by John Perry and Michael Bratman	
Particular Resource Req.:	None	
Teaching Strategy:	Instructor delivers lectures, conducts lab sessions, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.	

3.3.5 SPRT118 Moral and Civic Education

Course Code:	SPRT118
Course Title:	Moral and Civic Education
Prerequisites:	None
Credit Hours:	2 (4 ECTS)
Course Schedule:	Academic Year II
	Semester II
Description:	The course introduces learners to the latest debates on ideals and practices in national and moral education. Basic concepts related to national and moral education, such as family, morality and values, social ethics, nation and national identity, patriotism and citizenship, and their practice and development in both the domestic and international contexts will be examined. Through a reflective-inquiry approach, learners will be able to develop a critical understanding of the above concepts and theories.
Learning Outcomes:	Upon completion of this course, students will be able to: ● develop a reflective understanding of the fundamental concepts related to national and moral education, ● demonstrate an active, informed and responsible attitude in participating in the citizenship debates and investigating issues related to national and moral education; ● acquire basic reflective, inquiry and participation skills in taking part in debates and conducting investigations on issues related to national and moral education ● acquire a reflective understanding of the major social institutions and contexts for national and moral learning and education, especially their practice and development.

Course Content

Unit	Topics	Week
1	Understanding Civics and Ethics ● Defining Civic, Ethics and Morality ● Ethics and Law ● The importance of moral/civic education	1-2
2	Approaches to Ethics ● Normative Ethics ● Non-normative Ethics	3-5
3	Ethical decision making and moral judgement ● Making ethical decisions and actions ● Morality and Nature ● Individual Morality ● Being Morally and Ethically responsible	6-7
Mid Semester Week		8
4	State, Government and citizenship ● Understanding States ● Rival theories of State ● The role of states ● Understanding Government ● Understanding Citizenship	9-12
5	Constitution, Democracy and human rights ● Constitution and constitutionalism ● Constitutionalism	13-16

	● Constitutional Experience of Ethiopia pre and post 1931 ● Democracy and Democratization ● Human rights: Concepts and Theories	
Textbook and References:	1. Sage handbook of citizenship education and democracy. London, Sage. (2008). Arthur, J., Davies, I. and Hahn, C. (Eds.) 2. Moral and Civic Education Student Handbook prepared by Addis Ababa University	
Particular Resource Req.:	None	
Teaching Strategy:	Instructor delivers lectures, conducts tutorial sessions, prepares reading assignments and problems for group discussion, gives consultation and advises students on assignment solutions, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	Class participation: Participation in inquiries, discussions and debates conducted in the lessons (20%). A group research project with a written report on a national and moral education in Ethiopia (40%); Final Exam (40%)	

3.3.6 SPRT214 Social Anthropology

Course Code:	SPRT214
Course Title:	Social Anthropology
Prerequisites:	None
Credit Hours:	2 (4 ECTS)
Course Schedule:	Academic Year II
	Semester II
Description:	This course is expected to acquaint students with essential concept of anthropology covering a wide array of questions revolving around our very existence. It covers issues such as what makes human beings similar to each other? How do we differ from one another? What do anthropologist mean when they talk about diversity, multiculturalism, marginalization, inclusion and exclusion? The course enable learners grasp the different ways of being human by dealing with themes such as culture, kinship, marriage, cultural relativism, ethnocentrism, humanity, human origins, cosmologies, race, ethnicity, ethnic relations, ethnic boundaries, marginalization, minorities, local systems of governance, legal pluralism, indigenous knowledge systems, and indigenous practices and development.
Learning Outcomes:	Upon the successful completion of the course, students will be able to: ● Develop an understanding of the nature of anthropology ● Understand the cultural and biological diversity of humanity and unity in diversity across the world and in Ethiopia; ● Realize the socially constructed nature of identities & social categories such as gender, ethnicity, race and sexuality; ● Explore the various peoples and cultures of Ethiopia; ● Understand the social, cultural, political, religious& economic life of different ethno-linguistic & cultural groups of Ethiopia; ● Understand different forms marginalization and develop skills inclusiveness; ● Know about values, norms and cultural practices that maintain society together; and ● Develop broader views and skills to deal with people from a wide variety of socio-economic and cultural backgrounds.

Course Content

Unit	Topic	Week
1	Introducing Anthropology and its Subject Matter ● Definition, Scope and Subject Matter of Anthropology ● Sub-fields of anthropology ● Unique (Basic) Features of Anthropology ● Misconceptions about anthropology ● The Relationship between Anthropology and Other Disciplines ● The Contributions of anthropology	1-2
2	Human Culture and Ties that Connect ● Conceptualizing Culture: What Culture is and What Culture isn't ● Characteristic Features of Culture ● Aspects/Elements of Culture ● Cultural Unity and Variations: Universality, Generality and Particularity of Culture ● Evaluating Cultural Differences: Ethnocentrism, Cultural Relativism and Human Rights ● Culture Change	3-4

	● Ties That Connect: Marriage, Family and Kinship	
3	Human Diversity, Culture Areas and Contact in Ethiopia ● Human Beings & Being Human: What it is to be human? ● Origin of the Modern Human Species: Homo sapiens ● The Kinds of Humanity: human physical variation ● Human Races: the history of racial typing ● The Grand Illusion: Race, turns out, is arbitrary ● Why is Everyone Different? Human Cultural Diversity/Variation ● Culture area and cultural contact in Ethiopia	5-7
Mid Semester Week		8
4	Marginalized, Minorities, and Vulnerable Groups ● Definition of concepts ● Gender-based marginalization ● Marginalized occupational groups ● Age-based vulnerability ● Religious and ethnic minorities ● Human right approaches and inclusiveness: Anthropological perspectives	9-10
5	Identity, Inter-Ethnic Relations and Multiculturalism in Ethiopia ● Identity, Ethnicity and Race: Identification and Social Categorization ● Conceptualizing Ethnicity –What's it? ● Ethnic Groups and Ethnic Identity ● Race –The Social Construction of Racial Identity ● Theories of Ethnicity: Primordialism, Instrumentalism and Social Constructivism	11
6	Customary and Local Governance Systems and Peace Making ● Indigenous and local governance ● Intra and inter-ethnic conflict resolution institutions ● Inter-ethnic conflict resolution ● Women's role in conflict resolution and peace-making ● Legal pluralism: interrelations between customary, religious and state legal systems	12-13
7	Indigenous Knowledge Systems (IKS) and Practices ● Indigenous Knowledge Systems (IKS) ● Significance of indigenous knowledge ● Indigenous knowledge and development ● Preservation, Challenges and Limitations of IK	14-16
Textbook and References:	1. Social Anthropology Student Handbook prepared by Addis Ababa University, 2019 2. Anthropology and Social Theory: Culture, Power, and the Acting Subject (a John Hope Franklin Center Book). 2006 by Sherry B. Ortner 3. Introduction to Social Anthropology. 2016, Joy Hendry.	
Teaching Strategy:	The teacher or course facilitator who is assigned to deliver is recommended to make use of different active learning methods including: brainstorming, question and answer, group discussion, buzz-group, cross-over, home-works, reading assignments, peer teaching, and seldom active lecturing.	
Assessment:	To assess the progress of student, the instructor/ the course facilitator is expected to employ a continuous assessment technique in the form of quizzes, group and individual assignments, take-home exam, final exam, term paper.	

3.3.7 SPRT217 General Psychology

Course Code:	SPRT217
Course Title:	General Psychology
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year II
	Semester I
Description:	This course introduces students with the fundamental principles of psychology and to the major subjects of psychological inquiry. The course provides an introduction to the concepts and theories of psychology and to their application to real life situations. Topics include history, sensation, perception, consciousness, stress and coping, learning, memory, motivation and emotions. Basic concepts and principles of individual behaviour are examined, particularly those of human development, normal and abnormal behaviour, social psychology, learning, perception, and psychological measurement.
Learning Outcomes:	Upon completion of this course, students will be able to: ● Describe basic psychological concepts; ● Compare and contrast the major theoretical perspectives in psychology; ● Differentiate between scientific and non-scientific information about human behaviour and mental processes. ● Explain psychological processes involved in sensation, perception, learning, memory, motivation, emotion, states of consciousness and health. ● Analyze the variety of factors affecting sensation, perception, consciousness, learning, memory, motivation, emotion, and health. ● Apply psychological concepts and principles to situations in everyday life.

Course Content		
Unit	Topics	Week
1	Essence of Psychology ● Definition of Psychology and Related Concepts ● Goals of Psychology ● Historical Background and Major Perspectives in Psychology ● Branches/Sub Fields of Psychology ● Research Methods in Psychology	1-2
2	Human Development ● Basics of Human Development ● Principles of Human Development ● Aspects of Human Development ● Theories of Human Development	3-4
3	Learning and Theories Of Learning ● Definition, Characteristics and Principles of Learning ● Factors Influencing Learning ● Theories of Learning and their Applications	5
4	Memory and Forgetting ● Memory ● Forgetting ● Improving Memory	6
5	Motivation and Emotions ● Motivation ● Emotions	7
Mid Semester Week		8

6	Personality ● Meaning of Personality ● Theories of Personality	9
7	Psychological Disorders and Treatment Techniques ● Nature of Psychological Disorders ● Causes of Psychological Disorders (Based on Perspectives) ● Types of Psychological Disorders ● Treatment Techniques	10
8	Introduction to Life Skills ● Nature and Definition of Life skills ● Components and Goals of Life Skills ● Intra-personal and personal skills ● Self-Concept and Self-Awareness ● Self-Control and Anger Management ● Emotional Intelligence and Managing Emotion ● Stress, Coping with Stress and Resilience ● Critical and Creative Thinking; Problem Solving and Decision Making	11-12
9	Academic Skills ● Time Management ● Note-taking and Study Skills ● Test-Taking Skill ● Test Anxiety and Overcoming Test Anxiety ● Goal Setting; Career Development Skills	13-14
10	Social Skills ● Understanding cultural Diversity ● Gender and Social Inclusion ● Diversity Management ● Interpersonal Communication Skills ● Social Influences ● Peer Pressure ● Assertiveness ● Conflict and Conflict Resolution ● Team Work ● Overcoming Risky Behavior	15-16
Text book and Textbook and References:	1. Introduction to Psychology: Gateways to Mind and Behavior (MindTap Course List), 2018, by Dennis Coon and John O. Mitterer	
Teaching Strategy:	Instructor delivers lectures, conducts tutorial sessions, prepares reading assignments and problems for group discussion, gives consultation and advises students on assignment solutions, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 30%: midterm exam, 40%: Final Examination	

3.3.8 SPRT311 Business Accounting

Course Code:	SPRT311
Course Title:	Business Accounting
Prerequisites:	None

Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year III Semester I
Description:	The course has the general objective of introducing students to the basics of accounting and reporting of financial activities of business organizations. It is designed to provide introductory knowledge of accounting principles, concepts, and practices. The course deals with the processes involved in financial statements preparation for service-giving businesses, merchandising businesses, and manufacturing businesses. Topics covered includes the features of accounting information, users of accounting information, nature of financial statements, the double entry system, financial statements preparation process including journals, ledgers, trial balance, adjustments, worksheets, accruals, adjusting and closing entries, and the accounting system. The course provides a foundation for more advanced work in the fields of Accounting and business. All topics are studied in accordance with the provisions of International Financial Reporting Standards (IFRS)
Learning Outcomes:	Upon the successful completion of this course, the students will be expected to: ● Understand the role of Accounting in business and develop an awareness of the accounting profession ● Understand the purpose of the financial accounting function and standard financial accounting practices ● Define and apply accounting terminology, concepts, and principles ● Summarize and apply basic financial accounting terms, concepts, and principles. ● Take a series of transactions through the accounting cycle ● Analyze, record, and report transactions for service, merchandising, and manufacturing businesses. ● Apply accounting principles and control of cash and receivables

Course Content

Unit	Topics	Week
1	The context and purpose of financial reporting ● The reason for, and objectives of, financial reporting ● Users' and stakeholders'' needs ● The main elements of financial reports ● The regularity frame work ● The qualitative characteristics of financial reporting ● Alternative basses used in the preparation of financial information	1-2
2	The use of double entry and accounting system ● Double entry book keeping space principles including the maintenance of accounting records and source of information ● Ledger accounts, books of prime entry and journals ● Accounting systems and the impact of information technology on financial reporting	3-5

3	Recording transaction and events ● Sales and purchase ● Cash ● Inventory ● Tangible non-current assets and orientation ● Accruals and pre payments ● Receivables and payables ● Provisions and contingencies ● Capital structure and finance costs	6-9
Mid Semester Week		8
4	Preparing trial balance ● Trial balance ● Correction of errors ● Control accounts and reconciliations ● Bank reconciliations ● Suspense accounts	9-12
5	Preparing basic financial statements ● Statements of financial position ● Income statements and statement of comprehensive income ● Events affair the reporting period ● Accounting for partnership ● Statements of cash flows (excluding partnerships) ● Incomplete records	13-16
Textbook and References:	1. Business Accounting & Finance 6th edition Gowthorpe, Catherine (2024) 2. Financial Accounting: An Introduction, 9th edition by Pauline Weetman and Darren Jubb (2024)	
Resource Req.:	None	
Teaching Strategy:	Instructor delivers lectures, conducts lab sessions, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.	

3.3.9 SPRT312 Entrepreneurship

Course Code:	SPRT312
Course Title:	Entrepreneurship
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year III
	Semester II
Description:	This course is designed to introduce students to the concept of sustainable entrepreneurship, a manageable process that can be applied across careers and work settings. It focuses on building entrepreneurial attitudes and behaviors that will lead to creative solution within community and organizational environments. Course topics include the history of entrepreneurship, the role of entrepreneurs in the 21st century global economy, and the identification of entrepreneurial opportunities. The elements of creative problem solving, the development of a business concept/model, the examination of feasibility studies and the social /moral/ethical implication of entrepreneurship will be incorporated. Issues related to starting and financing a new venture are included.
Learning Outcomes:	Upon successful completion of the course, students will be able to: ● Define entrepreneurship within the context of society; organizations and individuals. ● Distinguish between an entrepreneurial and a conventional approach to management. ● Describe the element of an effective business model/plan. ● Develop a personal framework for managing the ethical dilemmas and social responsibilities facing entrepreneurs. ● Describe the leadership studies of entrepreneurs who have been successful in different sectors (e.g., start-ups, corporations. Community, public sector, etc.). ● Identify traits/characteristics of an entrepreneur/ entrepreneurs as exhibited in behavior. ● Analyze elements of the entrepreneurial mind set and discuss the implications for functioning as a successful entrepreneur. .

Course Content

Unit	Topics	Week
1	The Nature of Entrepreneurship ● Historical Origin of Entrepreneurship ● Definitions of Entrepreneurship and Entrepreneur ● Types of Entrepreneurs ● Role of Entrepreneurs in Economic Development ● Entrepreneurial Competence and Environment ● Creativity, Innovation and Entrepreneurship	1-2
2	Business Planning ● Opportunity Identification and Evaluation ● Business Idea Development ● Business Idea Identification ● Methods for Generating Business Ideas ● Concept of Business Plan ● Business plan Formats	3-4
	Business Formation ● The Concept of Small Business Development ● Forms of Business (A Short Explanation)	5-7

	● Definition and Role/Importance of SMEs in Developing Countries ● Setting up Small Scale Business ● Small Business Failure and Success Factors ● Classification of Enterprises in Ethiopian Context ● Problems of Small-Scale Business in Ethiopia ● Organizational Structure and Entrepreneurial Team Formation	
Mid Semester Week		8
	Product/Service Development ● The Concept of Product/Service Technology ● Product/Service Development Process ● Legal and Regulatory Frameworks for Entrepreneurs ● Intellectual Property Protection/Product/Service Protection ● The Intellectual Property System in Ethiopia	9-10
	Marketing ● Meaning and Definitions of Marketing ● Core Concepts of Marketing ● Importance of Marketing ● Marketing Philosophies ● Marketing Information Systems ● The Marketing Mix Strategy ● Selling and of Customer Service	11-12
	Business Financing ● Financial Requirements ● Sources of Financing ● Lease Financing; Traditional Financing in Ethiopian (Equib/Idir, Etc.) ● Crowd Funding ● Micro Finances	13-14
	Managing Growth and Transition ● Timmons Model of Entrepreneurship ● New Venture Expansion Strategies ● Business Ethics and Social Responsibility	15-16

Textbook and References:	1. Entrepreneurship: Theory, Process, and Practice (MindTap Course List), 2016, by Donald F. Kuratko 2. Entrepreneurship (Irwin Management), 2016 by Robert D Hisrich and Michael P Peters
Teaching Strategy:	Instructor delivers lectures, conducts lab sessions, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.

3.3.10 SPRT411 Inclusiveness

Course Code:	SPRT411
Course Title:	Inclusiveness
Prerequisites:	None
Credit Hours:	2 (4 ECTS)
Course Schedule:	Academic Year IV
	Semester I
Description:	This course intends to make students be more sensitive to the people they work with. Students will learn how to assess, understand and address the needs of persons with disabilities and vulnerabilities; and provide relevant support or seek extra support from experts. He/she also learns how to adapt and implement services for an inclusive environment that aims to develop holistic development such as affective, cognitive and psychosocial skills of the population with disabilities and vulnerabilities.
Learning Outcomes:	Upon successful completion of the course, students will be able to: ● Articulate personal attitudes, biases, and perspectives related to diversity in the workplace ● Identify the needs and potentials of persons with disabilities and vulnerabilities. ● Identify environmental and social barriers that hinder the needs, potentials and full participations, in all aspects of life of persons disabilities and vulnerabilities ● Demonstrate desirable inclusive attitude towards all persons with disabilities and vulnerabilities in full participations ● Apply various assessment strategies for service provisions for evidence-based planning and implementation to meet the needs of persons with disabilities and vulnerabilities · Adapt environments and services according to the need and potential of the persons with disabilities and vulnerabilities ● identify strategies to manage diversity issues within the workplace ● Determine and select strategies to ensure organizational inclusivity

Course Content

Unit	Topics	Week
1	Understanding Disabilities and Vulnerabilities ● Definitions of disability and vulnerability ● Types of disabilities and vulnerabilities ● Causes of disability and vulnerability ● Historical movements from segregation to inclusion ● The effects of attitude on the move towards inclusion	1-2
2	Concept of Inclusion ● Definition of Inclusion ● Principles of Inclusion ● Rationale for Inclusion ● Factors that Influenced Development of Inclusion ● Benefits of Inclusion ● Features of Inclusive Environment	3-4
3	Identification and Differentiated services ● Impact of Disability and Vulnerability on daily life ● Economic Factors and Disability	5-6

	● Political Factors and Disability ● Psychological Factors of Disability ● The family and disability ● Needs of Persons with Disabilities and Vulnerabilities. ● Gender and Disability	
4	Promoting Inclusive Culture ● Definition of Inclusive Culture ● Dimensions of Inclusive Culture ● Recruitment, Training, & Advancement Opportunities ● Workplace Accommodations and Accessibility ● Building Inclusive ● Means of establish inclusive culture ● Characteristics of an Inclusive organization	7-9
Mid Semester Week		8
5	Inclusion for Peace, Democracy and Development ● Definition of Peace, Democracy and Development ● Democratic principles of inclusive practices ● Inclusive Education for Development ● Respecting diverse needs, culture, values, demands and ideas ● Valuing diversity (cultural, ethnic, religion, etc.)	10-11
6	Legal Framework ● General Overview of Legal frameworks ● Legal Frameworks Regarding Inclusion ● The UN Conventions ● National Laws and Policy Frameworks	12
7	Resources Management for Inclusion ● Provision of Resources (Material, HR, etc) ● Accommodations ● Organization and Task Completion ● Collaborate partnership with stakeholders	13-14
Textbook and References:	1. Inclusiveness student handbook prepared by Addis Ababa University 2. Diversity Consciousness: Opening Our Minds to People, Cultures, and Opportunities" (4th Edition). 2015. by Richard D. Bucher; 3. An Inclusive Academy: Achieving Diversity and Excellence (The MIT Press) . 2018 by Abigail J. Stewart and Virginia Valian.	
Particular Resource Req.:	None	
Teaching Strategy:	Instructor delivers lectures, conducts lab sessions, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.	

3.3.11 SPRT416 History of Ethiopia and the Horn

Course Code:	SPRT416
Course Title:	History of Ethiopia and the Horn
Prerequisites:	None
Credit Hours:	3 (5 ECTS)
Course Schedule:	Academic Year IV
	Semester II
Description:	In this course students will learn about the role of history in human life and goals of studying history. They will also study the importance of history in nation building and the making of identity in time and space. The course covers the major historical processes in Ethiopia and the Horn from ancient times to 1994. It is also concerned with how the socio-cultural, religious, economic and political experiences of the past are interwoven in the making of contemporary Ethiopia and the Horn. It demonstrates how societies, peoples and the world that we live in have changed over time and its implication for the history of Ethiopia and the Horn. The course emphasizes the social, economic and cultural history of peoples in Ethiopia
Learning Outcomes:	After completing the course, students will be able to: ● distinguish the meaning, nature and uses of history. ● understand Ethiopia and the Horn in relation to Human Evolution and Neolithic Revolution ● trace origin, developments, achievements and effects of states in the region during the ancient period. ● realize the interplay between local developments and foreign influence in the making of the region. ● explicate the role of population movements in shaping modern Ethiopia and the Horn. ● assess developments in Eastern, Central, Southern & Western parts of Ethiopia & the Horn ● discern the move towards modernization and the challenges encountered ● elaborate the socio-economic and political changes of the post 1941 imperial period

Course Content

Unit	Topics	Week
1	Introduction ● Concepts of History: Meaning, Nature and Uses ● Sources & Methods of Historical Study ● Origin and Development of Historiography of Ethiopia and the Horn ● Introducing and Understanding Ethiopia and the Horn	1-2
2	Peoples and Cultures in Ethiopia and the Horn ● Human Evolution ● Neolithic Revolution ● The Peopling of the Region ● Religion and Religious Processes	3-4
3	Policies, Economy & Socio-Cultural Processes in Ethiopia & the Horn to the End of the 13th Century ● Evolution of States ● Ancient Polities ● External Contacts ● Economic Formations: Agriculture, Handicraft, Trade…	5-6

	● Socio-cultural achievements: Architecture, Writing, Calendar, Numerals…	
4	Politics, Economy & Socio-Cultural Processes from Late 13th–the beginning of the 16th Century ● "Restoration" of the "Solomonic'' Dynasty ● Power Struggle, Consolidation, Territorial and Religious Expansion of the Christian Kingdom Israel/"Falasha…" ● Social, Economic and Political Dynamics of Muslim Sultanates ● Rivalry between the Christian Kingdom and the Muslim Sultanates ● External Relations	7-9
Mid Semester Week		8
5	Politics, Economy & Socio-Cultural Processes from Early 16th–the End of the 18th Century ● Interaction and Conflicts of the Christian Kingdom and the Sultanate of Adal ● Foreign Interventions and Religious Controversies ● Population Movements of the Afar, Somali and Argobba ● Gadaa System and Oromo Population Movement (1522-1618) ● Interaction and Integration across Ethnic and Religious Diversities ● Peoples and States in Eastern, Central, Southern and Western Regions ● The Period of Gondar (1636-1769) and "Zemene Mesafint/Era of Princes" (1769-1855)	10-11
6	Internal Interactions and External Relations from the 1800–1941 ● The Nature of Interactions among peoples and states of Ethiopia and the Horn ● The Making of Modern Ethiopian State ● Socio-Economic Issues: agriculture, disease & famine, trade, slavery, manufacturing… ● External Relations, Challenges and Threats	12-13
7	Internal Interactions and External Relations from the 1941–1994 ● Post 1941 Imperial Period ● The *Derg* Regime (1974-1991) ● Historical Developments, 1991-1994	14-16
Textbook and References:	1. History of Ethiopia and the Horn. (Handout/reference prepared for students of higher learning in Ethiopia)	
Teaching Strategy:	Instructor delivers lectures, conducts lab sessions, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project term paper and 50%: Final Examination.	

3.3.12 SPRT418 Basics of Organizational Behaviour

Course Code:	SPRT418
Course Title:	Basics of Organizational Behaviour
Prerequisites:	None
Credit Hours:	2 (4 ECTS)
Course Schedule:	Academic Year IV
	Semester II
Description:	This 2-credit hour course provides an introduction to the key concepts of organizational behavior, focusing on individual and group dynamics within organizations. Topics include motivation, team dynamics, leadership, communication, and decision-making. Students will study how organizational culture and structure impact behavior and performance. Through case studies and practical examples, students will develop foundational skills to manage workplace relationships and enhance organizational effectiveness. The course is designed to equip students with the essential knowledge needed to address behavioral challenges and contribute to a positive work environment.
Learning Outcomes:	After completing the course, students will be able to: ● Define and explain fundamental concepts of organizational behavior, including motivation, team dynamics, leadership, communication, and decision-making. ● Assess how individual and group behaviors influence organizational effectiveness and performance. ● Identify and address common behavioral challenges in organizations through practical problem-solving techniques ● Demonstrate effective communication skills in both individual and group settings within an organizational context.

Course Content

Unit	Topic	Week
1	Introduction to Organizational Behavior ● Overview of Organizational Behaviour ● Importance of Studying Organizational Behavior ● Key Concepts and Terminology	1-2
2	Motivation and Personality ● In-Depth Exploration of Motivation Theories (e.g., Maslow's, Herzberg's) ● Personality Traits and Their Impact on Work Behavior ● Application of Personality Assessments	3-4
3	Team Dynamics and Leadership ● Stages of Team Development and Group Behavior ● Leadership Theories and Styles (e.g., Transformational, Transactional) ● Role of Leadership in Enhancing Team Performance	5-6
4	Communication and Decision-Making ● Communication Models and Barriers ● Decision-Making Models (e.g., Rational, Bounded Rationality) ● Strategies for Effective Communication and Problem-Solving	7
Mid Semester Week		8

5	Organizational Culture, Conflict, and Stress ● Understanding Organizational Culture and Structure ● Managing Conflict and Stress in the Workplace ● Strategies for Building Resilience and Handling organizational Change	9-10
6	Application and Review ● Case Studies on Organizational Behavior ● Integrating Concepts and Skills	11-13
7	Reflection and Presentations	14-16
Textbook and References:	1. Organizational behavior: Improving performance and commitment in the workplace (6th ed.)., 2018. By Colquitt, J. et al. 2. Organizational behavior: An introduction. 2016. by Cross, C., & Carbery, R. 3. Organizational behavior: An evidence-based approach, 2015 by Luthans, F. et al. 4. Organizational behaviour, 2016. By Robins, S. P., & Judge, T. A.	
Particular resource req.:	None	
Teaching strategy:	Instructor delivers lectures, conducts lab session, prepares reading assignments and topics for group discussion, prepares projects by discussion with student, gives consultation and advises students on project works and assignments, prepares and evaluates quiz, assignment, midterm and final examination.	
Assessment:	The evaluation shall be based on both formative and summative assessment which include: 30%: Continuous Assessment, 20%: Project and 50%: Final Examination.	

4. RESOURCES

The undergraduate program in Software Engineering is well organized in staff and teaching learning facilities. The general resources required for the program are summarized in the table below:

Resource	Description
Human Resource	● Eight full time Lecturers ● Eight Part time Lecturers ● Three graduate assistants / Technical Assistants
Classroom	● Four class rooms with LCD projector, Whiteboard and Internet connectivity
Computer Lab	● Two Computer Labs with 15 computers, capable of accommodating 25 students at a time
Staff Room	● Well-equipped room for academic and administration staff
Library	● Undergraduate library equipped with at least two reference materials (soft or hard copy) for each of the courses proposed in the curriculum
Software	● All required software is in place per the requirements of each course

.

5. Course Offering Schedule

5.1 Regular Program

Year I Semester I

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hou	Tutorial hours
SPRT111	College English I	3	5	48	0	0
SPRT117	Int. to Logic and Critical Thinking	3	5	48	0	0
MATH161	Discrete Mathematics	3	5	48	0	16
SWEN101	Introduction to Computer Systems	3	5	32	32	0
SPRT115	Geography of Ethiopia and the Horn	3	5	48	0	0
SWEN131	Fundamentals of Programming	3	5	32	32	0
	Total for the semester	18	30			

Year I Semester II

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SPRT112	College English II	3	5	48	0	0
SPRT118	Moral and Civic Education	2	4	32	0	0
SWEN104	Introduction to Software Engineering	3	5	48	0	0
SWEN132	Object Oriented Programming	4	7	48	32	0
ITSY154	Data Communication and Computer Networks I	3	5	32	32	0
MATH164	Linear Algebra	3	5	48	0	16
	Total for the semester	18	31			

Year II Semester I

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SPRT217	General Psychology	3	5	48	0	0
ITSY256	Information Assurance and Systems Security	2	4	32	32	0
SWEN241	Fundamentals of Database Systems	3	5	32	32	0
SWEN223	Software Requirements Engineering	3	5	48	0	0
SWEN233	Data Structures and Algorithms	3	5	32	32	0
MATH261	Calculus	3	5	48	0	16
	Total for the semester	17	29			

Year II Semester II

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SPRT214	Social Anthropology	2	4	32	0	0
SWEN232	Advanced Programming	4	7	48	32	0
SWEN224	Process Modeling and Workflow Design	3	5	48	0	0
SWEN226	Software Design and Architecture	4	7	48	0	0
SWEN252	Operating Systems	3	5	32	32	0
MATH266	Boolean Algebra	3	5	48	0	16

Total for the semester	19	33	

Year III Semester I

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SPRT311	Business Accounting and Management	3	5	48	0	0
SWEN381	Web Systems and Services	3	5	32	32	0
SWEN331	Mobile Application Development	3	5	32	32	0
SWEN327	Enterprise Systems	3	5	48	0	0
MATH361	Statistical Methods	3	5	48	0	16
SWEN363	Introduction to Artificial Intelligence	3	5	32	32	
	Total for the Semester	18	30			

Year III Semester II

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SPRT312	Entrepreneurship	3	5	48	0	0
SWEN322	Software Quality Assurance and Testing	3	5	48	0	0
SWEN324	Software Usability and Management	3	5	48	0	0
SWEN376	Software Project Management	3	5	48	0	0
SWEN366	Methods for IS Research	3	5	32	32	0
ITSY364	Foundations of Data Analytics	3	5	32	32	0
	Total for the Semester	18	30			

Year IV Semester I

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SPRT411	Inclusiveness	2	4	32	0	0
SWEN421	Software Process Improvement	3	5	48	0	0
SWEN423	Continuous Integration and Deployment	3	5	32	32	0
SWEN425	Service-Oriented Architecture	3	5	48	0	0
SWEN471	Systems Thinking and Systems Approach	3	5	48	0	0
	Elective I	3	5			
	Total for the Semester	17	29			

Year IV Semester II

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SWEN426	Seminar in Software Engineering	3	5	48	0	0
SWEN478	Software Product Management	3	5	48	0	0
SWEN492	Software Engineering Capstone Project I	4	7	64	0	0
SPRT416	History of Ethiopia and the Horn	3	5	48	0	0
SPRT418	Organizational Behavior	2	4	32	0	0
	Elective II	3	5	32	32	0
	Total for the Semester	18	31			

Note: Technical courses that do not require actual lab hours involve mandatory practical course work in real-life project setting under the supervision of instructors. In such courses, students are expected to use various tools in the computer lab for projects and assignments.

Year V Semester I

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SPRT591	Industry Practice	18	30	0	0	0
	Total for the Semester	18	30			

Year V Semester II

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SWEN522	Software Metrics	3	5	32	32	0
SWEN 524	Fundamentals of Financial Technology	**3**	**5**	48	0	0
SWEN552	Computer Simulation and Modelling	3	5	32	32	0
SWEN576	Management Information Systems	2	4	32	0	0
SWEN592	Software Engineering Capstone Project II	4	7	64	0	0
	Elective III	3	5	32	32	0
	Total for the Semester	18	31			

5.2 Extension Program

Year I Semester I

Code	Course Title	Credit hours	ECTS	Lec.hr.	Lab hou	Tutorial hours
SPRT111	College English I	3	5	48	0	0
SPRT115	Geography of Ethiopia and the Horn	3	5	48	0	0
MATH161	Discrete Mathematics	3	5	48	0	16
SWEN101	Introduction to Computer Systems	3	5	32	32	0
	Total for the semester	12	20			

Year I Semester II

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hou	Tutorial hours
SPRT112	College English II	3	5	48	0	0
SPRT117	Int. to Logic and Critical Thinking	3	5	48	0	0
SWEN131	Fundamentals of Programming	3	5	32	32	0
ITSY154	Data Communication and Computer Networks I	3	5	32	32	0
	Total for the semester	12	20			

Year I Summer

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
ITSY256	Information Assurance and Systems Security	2	4	32	32	0
SPRT217	General Psychology	3	5	48	0	0
	Total for the semester	5	9			

Year II Semester I

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SWEN104	Introduction to Software Engineering	3	5	48	0	0
SPRT118	Moral and Civic Education	2	4	32	0	0
SWEN132	Object Oriented Programming	4	7	48	32	0
MATH164	Linear Algebra	3	5	48	0	16
	Total for the semester	12	21			

Year II Semester II

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SWEN241	Fundamentals of Database Systems	3	5	32	32	0
SWEN223	Software Requirements Engineering	3	5	48	0	0
SPRT214	Social Anthropology	2	4	32	0	0
SWEN232	Advanced Programming	4	7	48	32	0
	Total for the semester	12	21			

Year II Summer

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SWEN252	Operating Systems	3	5	32	32	0
SWEN233	Data Structures and Algorithms	3	5	32	32	0
	Total for the semester	6	10			

Year III Semester I

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
MATH261	Calculus	3	5	48	0	16
SWEN224	Process Modeling and Workflow Design	3	5	48	0	0
SWEN226	Software Design and Architecture	4	7	48	0	0
	Total for the semester	10	17			

Year III Semester II

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SWEN331	Mobile Application Development	3	5	32	32	0
MATH266	Boolean Algebra	3	5	48	0	16
MATH361	Statistical Methods	3	5	48	0	16
SWEN363	Introduction to Artificial Intelligence	3	5	32	32	
	Total for the Semester	12	20			

Year III Summer

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SPRT311	Business Accounting and Management	3	5	48	0	0
SWEN366	Methods for IS Research	3	5	32	32	0
	Total for the Semester	6	10			

Year IV Semester I

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SWEN322	Software Quality Assurance and Testing	3	5	48	0	0
SWEN324	Software Usability and Management	3	5	48	0	0
SWEN381	Web Systems and Services	3	5	32	32	0
ITSY364	Foundations of Data Analytics	3	5	32	32	0
	Total for the Semester	12	20			

Year IV Semester II

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SWEN376	Software Project Management	3	5	48	0	0
SWEN421	Software Process Improvement	3	5	48	0	0
SWEN425	Service-Oriented Architecture	3	5	48	0	0
SWEN471	Systems Thinking and Systems Approach	3	5	48	0	0
	Total for the Semester	12	20			

Year IV Summer

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SWEN327	Enterprise Systems	3	5	48	0	0
SPRT312	Entrepreneurship	3	5	48	0	0
	Total for the Semester	6	10			

Year V Semester I

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SWEN478	Software Product Management	3	5	48	0	0
SWEN492	Software Engineering Capstone Project I	4	7	64	0	0
SPRT418	Organizational Behavior	2	4	32	0	0
	Elective I	3	5			
	Total for the Semester	12	21			

Year V Semester II

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SWEN522	Software Metrics	3	5	32	32	0
SWEN 524	Fundamentals of Financial Technology	3	5	32	32	0
SWEN576	Management Information Systems	2	4	32	0	0
	Elective II	3	5			
	Total for the Semester	11	19			

Year V Summer

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SWEN423	Continuous Integration and Deployment	3	5	32	32	0
SPRT416	History of Ethiopia and the Horn	3	5	48	0	0
	Total for the Semester	6	10			

Year VI Semester I

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SPRT591	Industry Practice	18	30	0	0	0
	Total for the Semester	18	30			

Year VI Semester II

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SWEN592	Software Engineering Capstone Project II	4	7	64	0	0
SWEN552	Computer Simulation and Modelling	3	5	32	32	0
	Elective III	3	5	32	32	0
	Total for the Semester	10	17			

Year VI Summer

Course Code	Course Title	Credit hours	ECTS	Lec. hours	Lab hours	Tutorial hours
SWEN426	Seminar in Software Engineering	3	5	48	0	0
SPRT411	Inclusiveness	2	4	32	0	0
	Total for the Semester	5	9			