

Ninevah University

جامعة نينوى



*First Cycle – Bachelor's Degree (B.Sc.) - College of
Information Technology – Software Dept.*

بكالوريوس – تكنولوجيا المعلومات - قسم البرمجيات



Table of Contents

1. Overview
2. Undergraduate Modules 2023-2024
3. Contact

1. Overview

This catalogue is about the courses (modules) given by the program of Information Technology to gain the Bachelor of software Science degree. The program delivers (40) Modules with (6000) total student workload hours and 240 total ECTS. The module delivery is based on the Bologna Process.

نظرة عامة

يتناول هذا الدليل المواد الدراسية التي يقدمها برنامج تكنولوجيا المعلومات للحصول على درجة بكالوريوس في علوم البرمجيات. يقدم البرنامج (40) مادة دراسية، مع (6000) إجمالي ساعات حمل الطالب و 240 إجمالي وحدات أوروبية. يعتمد تقديم المواد الدراسية على عملية بولونيا.

2. Undergraduate Courses 2023-2024

Module 1

Code	Course/Module Title	ECTS	Semester
NVU1101	Computer Skills	6	1
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
0	4	63	87
Description			
Computer skills refer to a person's ability to use and navigate computer systems, software, and hardware. Proficiency in computer skills is increasingly important in the modern world, where technology is integrated into most aspects of our lives. These skills can range from basic tasks, such as using word processing software, to more advanced skills like programming and data analysis. Fundamental computer skills include understanding operating systems and basic hardware components, creating and editing documents, handling email and web browsing, and using spreadsheets and database software. Advanced skills may include programming languages, networking, graphic design, and cyber security. Being proficient in computer skills can improve one's productivity, efficiency, and competitiveness in the job market.			

Module 2

Code	Course/Module Title	ECTS	Semester
CIT1120	Programming Principles I	6	1
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	2	63	87
Description			
<p>Programming Principles I is an introductory course that covers the fundamental concepts and principles of computer programming. This course is designed to teach students the basics of programming, including variables, data types, control structures, functions, and algorithms. By the end of the course, students will have gained experience in problem-solving and logical reasoning, as well as essential programming skills, such as debugging and testing. They will also develop an understanding of the importance of good coding practices and the importance of readability and maintainability of code. This course is an essential foundation for anyone interested in pursuing a career in computer science or software development.</p>			

Module 3

Code	Course/Module Title	ECTS	Semester
CIT1116	Calculus	6	1
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
3	1	63	87
Description			
<p>Calculus is a branch of mathematics that focuses on the study of rates of change and how things change over time. It is divided into two branches, differential calculus and integral calculus. Differential calculus deals with finding the instantaneous rate of change of a function, while integral calculus deals with calculating the total change of a function over a given interval. Calculus is used in a wide variety of fields, including physics, engineering, economics, and computer science, to name a few. The concepts of integral and differential calculus are the foundation of many advanced mathematical concepts and formulas, making it a crucial area of study for anyone interested in higher level mathematics or any field that requires advanced mathematical skills.</p>			

Module 4

Code	Course/Module Title	ECTS	Semester
CIT1115	Principles of Logic	7	1
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	2	63	112
Description			
<p>Principles of Logic is the study of reasoning and inference. Its principles are based on the use of language to express propositions and arguments, and the evaluation of the truth and validity of these propositions and arguments. These principles include the basic laws of propositional and predicate logic, as well as the rules of inference and deduction that are used to establish the validity of arguments. The principles of logic provide a framework for analyzing complex arguments and reasoning, and for ensuring that conclusions are logically sound and valid. The study of logic is fundamental to many fields, including mathematics, philosophy, computer science, linguistics, and law.</p>			

Module 5

Code	Course/Module Title	ECTS	Semester
NVU1104	English Language	5	1
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	2	62	63
Description			
<p>English is a West Germanic language that is spoken by over 1.5 billion people worldwide, making it the third most commonly spoken language after Mandarin and Spanish. It is the official language of many countries, including the United States, the United Kingdom, Australia, and Canada. English is known for its extensive vocabulary, complex grammar rules, and diverse dialects and accents. It is the language of international business, diplomacy, and academic discourse, and is widely used in entertainment, media, and popular culture. The study of English includes grammar, vocabulary, pronunciation, reading, writing, and speaking, and is essential for both personal and professional communication in today's globalized world. English is the language of international business, diplomacy, and academic discourse, and is essential for both personal and professional communication in today's globalized world. To give you an idea, basic vocabulary that forms part of college level English can be divided into categories such as: Nouns - people, places, things, and ideas. Verbs - action words used in sentences to convey meaning. Adjectives - words used to describe or qualify as nouns. Adverbs - words that modify verbs, adjectives, or other adverbs. Pronouns - words used in place of nouns Prepositions - words used to denote relationships between nouns. Conjunctions - words used to connect or join two clauses. Determiners - words used to identify nouns. Interjections - words that denote strong emotions .At advanced levels, there is a much wider range of vocabulary, including academic and technical terms.</p>			

Module 6

Code	Course/Module Title	ECTS	Semester
CIT1218	Statistics and Probabilities	5	2
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	1	47	78
Description			
<p>Statistics and Probabilities is the branch of mathematics concerned with collecting, analyzing, and interpreting data. It involves techniques such as graphical representation of data, descriptive statistics, inferential statistics, and hypothesis testing. Probability, on the other hand, is the study of uncertainty and randomness. It involves calculating the likelihood of an event occurring, based on prior knowledge and assumptions. Probability theory is used to analyze and model phenomena that exhibit random behavior, such as in physics, finance, and gambling. Statistics and probability are often used together to make informed decisions and predictions based on data and uncertainty. They are essential in many fields, including science, engineering, social sciences, and business.</p>			

Module 7

Code	Course/Module Title	ECTS	Semester
SOFT1220	Programming Principles II	8	2
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
3	3	93	107
Description			
<p>Programming Principles II is an advanced computer science course that builds on the foundational concepts and skills introduced in Programming Principles I. This course focuses on object-oriented programming paradigms and techniques, including abstraction, inheritance, encapsulation, polymorphism, and design patterns. Students learn how to design and implement complex software systems using languages such as Java, Python, or C++. The course also covers topics such as data structures, algorithms, and software development methodologies. Through hands-on projects and assignments, students gain practical experience in software design, coding, and testing, as well as problem-solving and critical thinking skills. Programming Principles II prepares students for advanced studies in computer science and professional careers as software developers, engineers, or architects.</p>			

Module 8

Code	Course/Module Title	ECTS	Semester
SOFT1215	Logic Design	7	2
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	3	78	97
Description			
<p>Logic Design is a fundamental computer science course that introduces students to the principles of digital logic and circuit design. This course covers Boolean algebra, logic gates, combinational and sequential circuits, and state machines. Students learn how to analyze, design, and optimize digital circuits using techniques such as Karnaugh maps and truth tables. They also gain hands-on experience with circuit simulation and implementation tools, and learn how to translate logical expressions into hardware designs. Logic Design lays the groundwork for understanding the operation of digital systems such as computers, microcontrollers, and programmable logic devices. This course is essential for students pursuing degrees or careers in computer engineering, electrical engineering, or related fields</p>			

Module 9

Code	Course/Module Title	ECTS	Semester
CIT1222	Computation Theory	6	2
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	1	47	103
Description			
<p>Computation Theory is a branch of computer science that studies the limits and capabilities of computation, as well as the design and implementation of algorithms and programming languages. It is concerned with topics such as formal languages, automata theory, computability, algorithmic complexity, and complexity theory. The theory explores the fundamental principles that govern the behavior of computational systems, including the basic building blocks of computation and the processes through which they interact. Computation Theory provides a framework for understanding the nature of computation and helps to guide the design and development of machines and software systems that can effectively solve complex problems. It is considered an important theoretical discipline with broad applications in computer science and related fields.</p>			

Module 10

Code	Course/Module Title	ECTS	Semester
CIT1225	Discrete Structures	4	2
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	1	47	53
Description			
<p>Discrete Structures is a branch of mathematics that focuses on discrete mathematical objects, such as integers, graphs, permutations, and sets. It deals with structures and relationships that can be represented and manipulated using logical and algebraic techniques. Discrete Structures provides the foundation for many areas of computer science, including algorithms, data structures, cryptography, and formal logic. It also has widespread applications in other fields, such as physics, chemistry, engineering, and economics. Topics in Discrete Structures include combinatorics, graph theory, Boolean algebra, set theory, and logic. Understanding of Discrete Structures is essential for anyone pursuing a career in computer science or related fields.</p>			

Module 11

Code	Course/Module Title	ECTS	Semester
SOFT2302	Data Structures and Algorithms I	6	3
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	3	78	72
Description			
<p>Data Structures and Algorithms I is a course that teaches fundamental concepts and techniques for organizing and processing data efficiently. It covers topics such as arrays, linked lists, stacks, queues, trees, and graphs, and explores various algorithmic approaches to sorting, searching, and manipulating these data structures. The course emphasizes algorithm analysis and design, including complexity analysis, asymptotic notation, and the use of recursion and dynamic programming to solve problems. Students learn how to implement data structures and algorithms in a programming language, and analyze their performance in terms of time and space complexity. This course is essential for computer science students who want to develop strong foundational skills in data management and algorithmic problem solving.</p>			

Module 12

Code	Course/Module Title	ECTS	Semester
SOFT2304	Object Oriented Programming (OOP)	6	3
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	3	78	72
Description			
<p>Object-oriented programming (OOP) is a programming paradigm that emphasizes the use of objects and their interactions to achieve a desired outcome. In OOP, programming is approached as a collection of objects that have properties and behaviors. Objects are instances of classes, which define their attributes, methods and relationships to other objects. Classes are used to create objects, and objects interact with one another through inheritance, interfaces, and polymorphism. OOP has several advantages, including improved code reusability, modularity, and ease of maintenance. By using OOP, developers can create software that is more scalable, flexible, and easier to understand and maintain over time, making it a popular choice for many modern software development projects.</p>			

Module 13

Code	Course/Module Title	ECTS	Semester
SOFT2305	Software Engineering (I)	7	3
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	3	78	97
Description			
<p>Software engineering is the process of designing, developing, testing, and maintaining software applications using a systematic approach. It involves the use of engineering principles, methods, and tools to create high-quality software that meets the needs of users. The software engineering process encompasses the entire software development life cycle (SDLC), from software requirements analysis to software design, implementation, testing, and maintenance. It also includes processes for project management, software configuration management, and quality assurance. Software engineering emphasizes the use of standardized processes and reusable components to reduce development time, improve software quality, and increase software reusability. Its main goal is to provide software that is reliable, efficient, and robust, while minimizing development costs and meeting the needs of users.</p>			

Module 14

Code	Course/Module Title	ECTS	Semester
SOFT2307	Software Systems	7	3
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	3	78	97
Description			
<p>Software systems are complex applications that enable users to accomplish specific tasks or goals. These systems are designed to process data, automate business processes, perform calculations, and manage information. Software systems can range from individual applications that perform basic tasks, such as word processing or spreadsheets, to large enterprise-wide applications that support complex business processes. Software systems are designed using various software development methodologies and programming languages, and are typically built using a layered architecture that separates the presentation layer, business logic layer, and data storage layer. The development of software systems requires a team of developers, testers, and project managers to work together to ensure that the software meets requirements, is scalable, reliable, and easy to maintain.</p>			

Module 15

Code	Course/Module Title	ECTS	Semester
NVU2310	English Language	4	3
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	1	47	53
Description			
<p>English is a West Germanic language that is spoken by over 1.5 billion people worldwide, making it the third most commonly spoken language after Mandarin and Spanish. It is the official language of many countries, including the United States, the United Kingdom, Australia, and Canada. English is known for its extensive vocabulary, complex grammar rules, and diverse dialects and accents. It is the language of international business, diplomacy, and academic discourse, and is widely used in entertainment, media, and popular culture. The study of English includes grammar, vocabulary, pronunciation, reading, writing, and speaking, and is essential for both personal and professional communication in today's globalized world. English is the language of international business, diplomacy, and academic discourse, and is essential for both personal and professional communication in today's globalized world. To give you an idea, basic vocabulary that forms part of college level English can be divided into categories such as: Nouns - people, places, things, and ideas. Verbs - action words used in sentences to convey meaning. Adjectives - words used to describe or qualify as nouns. Adverbs - words that modify verbs, adjectives, or other adverbs. Pronouns - words used in place of nouns. Prepositions - words used to denote relationships between nouns. Conjunctions - words used to connect or join two clauses. Determiners - words used to identify nouns. Interjections - words that denote strong emotions. At advanced levels, there is a much wider range of vocabulary, including academic and technical terms.</p>			

Module 16

Code	Course/Module Title	ECTS	Semester
SOFT2402	Data Structures and Algorithms II	6	4
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	3	78	72
Description			
<p>Data Structures and Algorithms II is a course that builds on the fundamentals of data structures and algorithm analysis. In this course, students are exposed to more advanced data structures such as trees, graphs, heaps, and hash tables, and the advanced algorithms that operate on them. The course covers topics such as dynamic programming, advanced searching and sorting algorithms, greedy algorithms, and graph algorithms. Students also learn about complexity theory and how to analyze the efficiency of algorithms in terms of time and space complexity. The course is designed to provide students with the skills and knowledge required to design and analyze efficient algorithms for solving complex problems in computer science and related fields.</p>			

Module 17

Code	Course/Module Title	ECTS	Semester
SOFT2406	Database Management Systems	6	4
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	3	78	72
Description			
<p>Database Management Systems (DBMS) is a course that provides a comprehensive overview of the principles, design, implementation, and management of database systems. In this course, students learn about the different types of data models, relational algebra, normalization, and data integrity. Additionally, students are taught how to design and implement a database using various software tools, such as SQL, and how to properly manage a database system. The course also covers the topics of transaction management, concurrency control, data warehousing, and data mining. Overall, this course equips students with the necessary skills and knowledge to create and manage robust database systems that meet the needs of organizations and businesses.</p>			

Module 18

Code	Course/Module Title	ECTS	Semester
SOFT2416	Software Engineering (II)	7	4
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	3	78	97
Description			
<p>Software Engineering (II) is an advanced course that builds on the foundational principles learned in Software Engineering (I). The course focuses on the development of large-scale software projects and explores various methodologies, techniques, and tools used in the software engineering process. Topics covered include software requirements analysis, design patterns, architectural styles, software testing and quality assurance, software maintenance, and project management. Students learn how to collaborate effectively in team environments, use industry-standard software development tools, and apply agile development methodologies to create robust software solutions that meet the needs of stakeholders. This course is essential for computer science students who want to become skilled software engineers capable of delivering high-quality software projects on time and within budget.</p>			

Module 19

Code	Course/Module Title	ECTS	Semester
SOFT3427	Software Requirements	7	4
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	2	62	113
Description			
<p>Software requirements describe the functional and non-functional capabilities that a piece of software must possess to meet the needs of its intended users. These requirements define the scope of the software project and help developers understand what they need to build. They typically include information about the operating system, hardware, and software components that are necessary for the software to operate effectively, as well as any integration requirements for other systems. Additionally, software requirements encompass various quality characteristics, like security, performance, maintainability, and usability. Overall, it is important to establish clear and comprehensive software requirements to ensure that the final product reliably delivers the intended functionality and meets the needs of its users.</p>			

Module 20

Code	Course/Module Title	ECTS	Semester
NVU2405	Human Rights	4	4
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	0	32	68
Description			
<p>حقوق الإنسان هي مجموعة من الحقوق الأساسية التي يتمتع بها كل إنسان، من دون تمييز أو إقصاء أو تحيز أو تمييز على أساس الجنس أو اللون أو العرق أو الدين أو المعتقد أو اللغة أو الوضع الاجتماعي أو السلالة أو الثروة أو الولاء السياسي أو أي وضع آخر.</p> <p>تشمل حقوق الإنسان العديد من الحقوق، بما في ذلك حقوق المدنية والسياسية، مثل حق الحياة والحرية والاجتماع والتعبير والدين والمعتقد والمساواة أمام القانون. وتشمل أيضًا حقوق الإنسان الاقتصادية والاجتماعية والثقافية، مثل حق العمل والتعليم والصحة والسكن والأمن الاجتماعي والثقافة.</p> <p>تعتبر حقوق الإنسان جزءًا من الكرامة الإنسانية بشكل عام، وهي تهدف إلى تحقيق المساواة والعدالة والحرية والكرامة لجميع البشر، وتعتبر حماية حقوق الإنسان وتطويرها أهم أولويات المجتمعات الإنسانية في جميع أنحاء العالم. وتحقيق حقوق الإنسان يعتبر ضروريًا للحفاظ على السلم والأمن العالميين، وانتشار العدالة والديمقراطية والتعايش السلمي في المجتمعات المختلفة.</p>			

Module 21

Code	Course/Module Title	ECTS	Semester
SOFT3556	AI Techniques	7	5
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	3	78	97
Description			
<p>AI Techniques is a course that delves into various methods and approaches used for developing and applying artificial intelligence (AI) technologies. Students explore different AI techniques such as machine learning, natural language processing, expert systems, and robotics. They learn about the applications of these techniques across different industries and domains, such as healthcare, finance, and transportation. The course also covers ethical considerations and challenges related to the development and deployment of AI technologies. Students gain a foundational understanding of AI concepts, tools, and applications, and use this knowledge to write code and implement AI techniques in real-world scenarios.</p>			

Module 22

Code	Course/Module Title	ECTS	Semester
SOFT3526	Compilers	7	5
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	3	78	97
Description			
<p>Compilers is a course that focuses on the theory and practice of compiler construction. The course covers the different phases of compilation, including lexical analysis, parsing, code generation, and optimization. Students learn about the design principles and algorithms used in compilers, as well as how to apply these concepts to develop efficient and reliable compilers for different programming languages. They also explore advanced topics such as compiler optimization and code generation for modern architectures. The course provides students with the necessary skills and knowledge to design and implement compilers, and to understand the inner workings of programming languages and their associated tools.</p>			

Module 23

Code	Course/Module Title	ECTS	Semester
SOFT3531	Data Communications and Networking	6	5
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	2	63	87
Description			
<p>Data Communications and Networking is a course that covers the principles of transmitting and receiving data between computer systems. The course covers various topics such as computer network architectures, network protocols, network security, and data transmission technologies. Students learn about the OSI and TCP/IP network models, LANs, WANs, and wireless networking, as well as Internet protocols such as TCP, UDP, IP, and DNS. The course covers various technologies such as Ethernet, Wi-Fi, Bluetooth, and cellular networks, and their respective protocols. The goal of the course is to provide students with the necessary knowledge and skills to design, implement, and troubleshoot computer networks and to understand the underlying principles of communication technologies.</p>			

Module 24

Code	Course/Module Title	ECTS	Semester
SOFT3522	Computer Architecture	6	5
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	2	63	87
Description			
<p>Computer architecture refers to the conceptual design and fundamental operational structure of a computer system. It describes how computer hardware is organized and how the various components interact with each other to perform complex tasks. Computer architecture involves the design of both the internal hardware components, such as the CPU, memory, and storage, as well as the external interfaces, such as input/output devices and network adapters. The goal of computer architecture is to create a system that is efficient, reliable, and flexible and can execute a wide range of software applications. Advances in computer architecture have driven advancements in computing power, speed, and functionality, enabling the development of powerful technologies such as machine learning, artificial intelligence, and big data analysis.</p>			

Module 25

Code	Course/Module Title	ECTS	Semester
NVU3510	English Language III	4	5
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	2	62	38
Description			
<p>English is the language of international business, diplomacy, and academic discourse, and is essential for both personal and professional communication in today's globalized world. To give you an idea, basic vocabulary that forms part of college level English can be divided into categories such as: Nouns - people, places, things, and ideas. Verbs - action words used in sentences to convey meaning. Adjectives - words used to describe or qualify as nouns. Adverbs - words that modify verbs, adjectives, or other adverbs. Pronouns - words used in place of nouns. Prepositions - words used to denote relationships between nouns. Conjunctions - words used to connect or join two clauses. Determiners - words used to identify nouns. Interjections - words that denote strong emotions. At advanced levels, there is a much wider range of vocabulary, including academic and technical terms</p>			

Module 26

Code	Course/Module Title	ECTS	Semester
SOFT3620	Operating System	7	6
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	3	78	97
Description			
<p>An operating system (OS) is a crucial set of software programs that controls, manages, and coordinates the hardware and software resources of a computer system. Its primary function is to provide a stable and efficient platform for users to run their programs and applications. Common examples of operating systems include Windows, macOS, and Linux. They perform critical tasks such as memory management, process scheduling, and security management. An operating system also provides a user-friendly interface for users to interact with their computer systems, including launching applications, managing files and folders, and connecting to the internet. Overall, operating systems are essential to computer systems' functioning, laying the foundation for users to perform their day-to-day tasks with ease and efficiency.</p>			

Module 27

Code	Course/Module Title	ECTS	Semester
SOFT3629	Software Engineering Tools	6	6
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	3	78	72
Description			
<p>Software engineering tools are applications or platforms that assist developers and engineers in creating, testing, and maintaining software programs. These tools include a wide range of software products such as Integrated Development Environments (IDEs), debuggers, compilers, testing suites, and version control systems. IDEs are the primary tools used to write and edit code, and they often include code completion and debugging features. Debuggers help find and fix bugs in the code, while compilers translate code into executable programs. Testing suites help automate testing processes, and version control systems help ensure that changes to the code are tracked and easily reversible. Together, these tools enable software engineers to create high-quality, reliable software programs.</p>			

Module 28

Code	Course/Module Title	ECTS	Semester
SOFT3623	Web Engineering	7	6
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	3	78	97
Description			
<p>Web Engineering is the application of software engineering principles to the design, development, and maintenance of web-based applications. It involves the use of various technologies, frameworks, and development methodologies to create high-quality, scalable, and secure web systems. Web Engineering includes a wide range of activities, including requirements gathering, system architecture design, web programming, testing, documentation, and maintenance. The goal of Web Engineering is to develop web applications that effectively meet the needs of users, are easy to use, and provide a seamless user experience. The field of Web Engineering has become increasingly important in recent years, as more and more businesses and organizations rely on web-based applications to provide products and services to their customers.</p>			

Module 29

Code	Course/Module Title	ECTS	Semester
SOFT3630	Software Quality and Reliability	6	6
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	2	63	87
Description			
<p>Software quality and reliability refer to the ability of software systems to perform their intended functions in a consistent and dependable manner, without errors or unexpected failures. Achieving high levels of software quality and reliability requires a combination of effective software development practices, rigorous testing and verification procedures, and ongoing maintenance and support. Common techniques used to assess and enhance software quality and reliability include code reviews, testing frameworks, fault-tolerant design principles, performance monitoring, and system robustness testing. Ultimately, software quality and reliability are critical factors in ensuring that software systems meet user needs, are scalable and maintainable, and are able to deliver consistent and reliable performance over time.</p>			

Module 30

Code	Course/Module Title	ECTS	Semester
NVU3607	Research methods	4	6
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	0	32	68
Description			
<p>Research methods refer to the techniques and procedures used to carry out a scientific investigation or study. There are various research methods employed, depending on the nature of the phenomena being investigated, the research objectives, and data collection requirements. Some commonly used research methods include experimental research, survey research, case study research, qualitative research, and quantitative research. Each research method has its own strengths and limitations, and the selection of a particular research method depends on the research question and the data analysis requirements. Through the use of rigorous research methods, researchers can produce reliable and valid findings that advance scientific knowledge and understanding.</p>			

Module 31

Code	Course/Module Title	ECTS	Semester
SOFT4720	Information Security	6	7
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	2	63	87
Description			
<p>Information security is the practice of protecting information and data from unauthorized access, use, disclosure, disruption, modification, or destruction. This involves implementing measures to ensure the confidentiality, integrity, and availability of information and data. Information security is critical for the protection of sensitive and personal information, financial assets, and intellectual property. Some techniques used to achieve information security include access controls, encryption, firewalls, intrusion detection systems, and security audits. Effective information security requires a combination of technological, organizational, and procedural measures, as well as education and awareness among employees and stakeholders. Failure to implement effective information security measures can result in serious consequences, such as financial loss, reputation damage, and legal liability.</p>			

Module 32

Code	Course/Module Title	ECTS	Semester
SOFT4724	Mobile Applications	7	7
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	2	63	112
Description			
<p>Mobile applications, also known as mobile apps, are software applications designed for use on mobile devices such as smartphones, tablets, and wearable devices. These apps are developed and deployed for specific mobile operating systems such as Android or iOS. Mobile applications usually provide users with access to specific functions, features, and content, such as communication, entertainment, productivity, and utility. They are designed to be intuitive, responsive, and efficient, often utilizing device-specific functionalities like cameras, GPS, and sensors, to enhance user experience and app functionality. With the global shift towards mobile devices, Mobile Applications have become a crucial aspect of modern communication and technology-driven lifestyles.</p>			

Module 33

Code	Course/Module Title	ECTS	Semester
SOFT4731	Reverse Engineering	7	7
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	2	63	112
Description			
<p>Reverse engineering is the process of analyzing a product, device, or software to understand its design, functionality, and components. The goal of reverse engineering is to extract useful information that can be used to improve, modify, or replicate the original product. This technique is commonly used in computer software engineering, mechanical engineering, and electrical engineering. The process can involve disassembling a product to identify its individual parts or examining the software code to understand its logic and functionality. Reverse engineering is a useful tool for improving existing products, creating new products, and understanding the competition. However, reverse engineering is also used for malicious purposes, such as creating counterfeit products or stealing intellectual property. In some cases, reverse engineering may also be illegal or violate intellectual property rights.</p>			

Module 34

Code	Course/Module Title	ECTS	Semester
SOFT4700	Graduation Project I	6	7
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
0	1	17	133
Description			
<p>A software graduation project is a research-based software application that is developed by a student or a group of students as part of their graduation requirements. The project typically involves identifying a problem or a need within a specific field and developing a software application to address it. This type of project allows students to apply their knowledge and skills in software development and gain practical experience in working on a real-world project. The goal of the software graduation project is to demonstrate the student's ability to design, implement, and test a complex software application that meets specific requirements. The outcome of the project can be a valuable addition to the student's portfolio and increase their employability in the software development industry.</p>			

Module 35

Code	Course/Module Title	ECTS	Semester
NVU4710	English Language IV	4	7
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	2	62	38
Description			
<p>English is the language of international business, diplomacy, and academic discourse, and is essential for both personal and professional communication in today's globalized world. To give you an idea, basic vocabulary that forms part of college level English can be divided into categories such as: Nouns - people, places, things, and ideas. Verbs - action words used in sentences to convey meaning. Adjectives - words used to describe or qualify as nouns. Adverbs - words that modify verbs, adjectives, or other adverbs. Pronouns - words used in place of nouns. Prepositions - words used to denote relationships between nouns. Conjunctions - words used to connect or join two clauses. Determiners - words used to identify nouns. Interjections - words that denote strong emotions. At advanced levels, there is a much wider range of vocabulary, including academic and technical terms</p>			

Module 36

Code	Course/Module Title	ECTS	Semester
SOFT4802	Network Security	6	8
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	2	63	87
Description			
<p>Network security is the process of protecting computer networks from unauthorized access, hacking, and malicious attacks. It comprises different technologies, such as firewalls, intrusion detection and prevention systems (IDS/IPS), antivirus software, and virtual private networks (VPNs), all aimed at safeguarding the network infrastructure and the information stored on it. Network security strategies may include implementing access controls to limit who can access the network and monitoring the network traffic to identify any unauthorized activities. The primary goal of network security is to ensure confidentiality, integrity, and availability of data and resources. A comprehensive network security approach can help prevent cyber attacks, data breaches, and other security incidents.</p>			

Module 37

Code	Course/Module Title	ECTS	Semester
SOFT4832	Software Development Models	6	8
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	2	78	72
Description			
<p>Software development models are frameworks that describe the different activities and tasks involved in developing software products. A software development model mainly covers the overall software development process, from planning, designing, developing, testing, deploying, and maintaining software applications. There are different software development models, including the Waterfall model, Agile model, Spiral model, and V-Model, each with different approaches and characteristics. The Waterfall model involves a sequential, linear process where one phase of development is completed before moving onto the next phase. The Agile model emphasizes continuous software delivery, flexible planning, and an iterative approach that emphasizes customer feedback. The Spiral model incorporates elements of both Waterfall and Agile models in an iterative approach that emphasizes risk management. The V-Model aligns testing with development by incorporating the quality assurance activities at each phase of the software development life cycle.</p>			

Module 38

Code	Course/Module Title	ECTS	Semester
SOFT4833	Software Maintenance	6	8
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	2	63	87
Description			
<p>Software maintenance refers to the process of making updates and modifications to an existing software application or system after the software has been deployed. This includes activities that are performed to fix any defects, add new features or functionality, or enhance the system's performance. There are different types of software maintenance, including corrective maintenance to fix errors, adaptive maintenance to modify the software due to changes in the environment, perfective maintenance to improve performance, and preventive maintenance to improve the software's maintainability and reliability. Software maintenance is essential to ensure that the software continues to function properly and meets the changing needs of the users and the business over time. It involves a range of activities performed by developers, testers, and support staff to ensure the software products are up-to-date, reliable, and secure.</p>			

Module 39

Code	Course/Module Title	ECTS	Semester
SOFT4834	Open Source Software	6	8
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
2	2	78	72
Description			
<p>Open source software is a type of computer software that is developed collaboratively and made available to the public to use, modify and distribute for free. This means that the source code is available for anyone to access and modify according to their own requirements. Open source software encourages collaboration, creativity and innovation by enabling developers to build upon the existing code to create new applications or improve functionality. One of the main benefits of open-source software is that it can be freely distributed, meaning that it can help to reduce costs associated with developing and deploying software. Another advantage of open-source software is that it allows people to work together and create unique solutions to complex problems.</p>			

Module 40

Code	Course/Module Title	ECTS	Semester
SOFT4800	Graduation Project II	6	8
Class (hr/w)	Lect/Lab./Prac./Tutor	SSWL (hr/sem)	USWL (hr/w)
	1	17	133
Description			
<p>Graduation Project II in software is a computer program that is developed by students in their final year of college as part of their graduation requirement. It is usually a project that involves designing, implementing and developing a software system to solve a specific problem or meet a particular need. The software is typically developed using programming languages, tools and technologies that are relevant to the problem domain. The software project may involve several stages such as requirements gathering, design, implementation, testing and deployment. The goal of Graduation Project II software is to provide students with an opportunity to apply the knowledge and skills they have acquired throughout their studies to create a software solution that meets specific criteria and requirements.</p>			

Contact

Program Manager:

Azhar Sabah Abdulaziz | Ph.D. in Computer Engineering | lecturer.

Email: azhar.abdulaziz@uoninevah.edu.iq

Mobile no.: +9647730135332

Program Coordinator:

Mostafa Ismael Shukri | Master in Software Engineering | lecturer.A

Email: mostafa.i.windi@uoninevah.edu.iq

Mobile no.:+9647705998814
