Ministry of Higher Education and Scientific Research
Scientific Supervision and Scientific Evaluation Apparatus
Directorate of Quality Assurance and Academic Accreditation
Accreditation Department

Academic Program and Course Description Guide

2024-2025

Introduction:

The educational program is a well-planned set of courses that include procedures and experiences arranged in the form of an academic syllabus. Its main goal is to improve and build graduates' skills so they are ready for the job market. The program is reviewed and evaluated every year through internal or external audit procedures and programs like the External Examiner Program.

The academic program description is a short summary of the main features of the program and its courses. It shows what skills students are working to develop based on the program's goals. This description is very important because it is the main part of getting the program accredited, and it is written by the teaching staff together under the supervision of scientific committees in the scientific departments.

This guide, in its second version, includes a description of the academic program after updating the subjects and paragraphs of the previous guide in light of the updates and developments of the educational system in Iraq, which included the description of the academic program in its traditional form (annual, quarterly), as well as the adoption of the academic program description circulated according to the letter of the Department of Studies T 3/2906 on 3/5/2023 regarding the programs that adopt the Bologna Process as the basis for their work.

In this regard, we can only emphasize the importance of writing an academic programs and course description to ensure the proper functioning of the educational process.

Concepts and terminology:

Academic Program Description: The academic program description provides a brief summary of its vision, mission and objectives, including an accurate description of the targeted learning outcomes according to specific learning strategies.

Course Description: Provides a brief summary of the most important characteristics of the course and the learning outcomes expected of the students to achieve, proving whether they have made the most of the available learning opportunities. It is derived from the program description.

Program Vision: An ambitious picture for the future of the academic program to be sophisticated, inspiring, stimulating, realistic and applicable.

Program Mission: Briefly outlines the objectives and activities necessary to achieve them and defines the program's development paths and directions.

Program Objectives: They are statements that describe what the academic program intends to achieve within a specific period of time and are measurable and observable.

Curriculum Structure: All courses / subjects included in the academic program according to the approved learning system (quarterly, annual, Bologna Process) whether it is a requirement (ministry, university, college and scientific department) with the number of credit hours.

Learning Outcomes: A compatible set of knowledge, skills and values acquired by students after the successful completion of the academic program and must determine the learning outcomes of each course in a way that achieves the objectives of the program.

Teaching and learning strategies: They are the strategies used by the faculty members to develop students' teaching and learning, and they are plans that are followed to reach the learning goals. They describe all classroom and extra-curricular activities to achieve the learning outcomes of the program.

Academic Program Description Form

University Name: wasit university

Faculty/Institute: college of computer science and information technology

Scientific Department: software department

Academic or Professional Program Name:

Final Certificate Name:

Academic System:

Description Preparation Date:

File Completion Date:

Assist. Prof. Dr.
Abdul Hadi M. Alaidi
Dean Assistant for
Scientific Affairs

Signature:

Head of Department Name:

الدكتور
احمد رعد عبد الحسين
رئيس قسم البرامجيات

Date: 16/2/2025

Signature:

Scientific Associate Name:

Date: 13/2/2025

The file is checked by:

Department of Quality Assurance and University Performance

Director of the Quality Assurance and University Performance Department:

Date: 10/2/2025

Signature:

Asst. Prof. Dr
Sait Ali Alsaidi
Dean college of computer
scenic & Information Technology

Approval of the Dean

13-2-2025

Course Description Form

Course Name
Research Methodology
Course Code
Semester / Year
2/ 2024-2025
The date of preparation this description
9/2/2025
Available Attendance Forms
Classrooms
Number of credit hours (total) / number of units (total)
2/2 ....
The name of the course administrator (if more than one name is mentioned)
Name: Assist.Prof.Dr. Riyadh Rahef Nuiaa Al Ogaili **Email: riyadh@uowasit.edu.iq**
8. Course Objectives

Course Objectives	• Understand the Research Process • Develop Research Questions and Design Studies • Conduct Comprehensive Literature Reviews • Adhere to Ethical Research Practices • Gain Proficiency in LaTeX • Communicate and Disseminate Research • Engage in Independent Research

9. Teaching and Learning Strategies

Strategy	• Interactive Lectures • Hands-On Lab Sessions • Flipped Classroom • Collaborative Group Work • Peer Review and Feedback • Case Studies and Real-World Examples • Scaffolded Assignments • Use of Digital Tools and Online Platforms • Continuous Assessment and Formative Feedback • Guest Lectures and Workshops

10. Course Structure

Evaluation method	Learning method	Unit or subject name	Required Learning Outcomes	Hours	Week
		Introduction to Research and LaTeX	What Is Research Types of Research The Research Process Introduction to LaTeX Setting Up a LaTeX Environment Practical Activity	2	1
		Research Paradigms and LaTeX Templates	Research Paradigms Qualitative vs. Quantitative Methods LaTeX Templates Practical Activity	2	2
		Research Ethics and LaTeX Formatting	Research Ethics LaTeX Formatting Basics Practical Activity	2	3
		Literature Review and LaTeX Bibliographies	Conducting a Literature Review Writing a Literature Review LaTeX Bibliographies with BibTeX Practical Activity	2	4
		Research Design and LaTeX Equations	Formulating Research Questions and Hypotheses Designing Research Methodologies LaTeX for Equations and Figures Practical Activity	2	5
		Data Collection and LaTeX Tables	Data Collection Methods LaTeX Tables Practical Activity	2	6
		Data Analysis Techniques and LaTeX Algorithms	Data Analysis Methods Presenting Algorithms and Pseudocode in LaTeX	2	7

			Practical Activity		
		Research Proposal Writing and LaTeX Document Structure	Components of a Research Proposal Writing a Compelling Proposal LaTeX Document Structure Practical Activity	2	8
		Case Studies in Software Research and LaTeX Examples	Analyzing Published Research Implications for Practice LaTeX Code Examples Practical Activity	2	9
		Presentation Skills and LaTeX Beamer	Effective Research Presentations Using LaTeX Beamer Practical Activity	2	10
		Research Dissemination and Scientific Writing	Publishing Research Findings Scientific Writing Best Practices Practical Activity	2	11
		Research Supervision and Mentorship	Role of Research Supervisors Building Effective Mentorship Integrating Feedback Practical Activity	2	12
		Independent Research Project Development	Project Planning and Development Ongoing Supervisor Meetings Practical Activity	2	13
		Final Research Project Presentations and Reflections	Research Project Presentations Class Discussion and Reflection Course Evaluation Practical Activity	2	14

11. Course Evaluation

12. Teaching and learning resources

Research Design: Qualitative, Quantitative, and Mixed Methods Approaches (5th Edition)
Authors: John W. Creswell & J. David Creswell (2022)

Course Description Form

Course Name
Software Architecture
Course Code
SWA
Semester / Year
2/ 2024-2025
The date of preparation this description
01/03/2025
Available Attendance Forms
Classrooms
Number of credit hours (total) / number of units (total)
45 hours– 3 units ....
The name of the course administrator (if more than one name is mentioned)
Name: Assit.Prof.Dr.Ahmad Shaker Abdalrada **Email:** ***aabdalra@uowasit.edu.iq***
8. Course Objectives

Course Objectives	**How to handle software's developments**

9. Teaching and Learning Strategies

Strategy	**A1- Introduce the principles and fundamentals of software architecture and its types.** **A2- Apply software architecture concepts.** **A3- Recognize the importance of software architecture.** **A4- Develop the ability to describe software architecture.** **Skill-Based Objectives:** **B1- Ability to deliver a lecture on software architecture.** **B2- Ability to design software architecture.** **B3- Ability to learn and train on various software architecture systems.** **B4- Ability to manage discussions and debates related to software architecture.** **Teaching and Learning Methods:** **Utilizing technology in lectures.** **Writing computer programs related to software architecture.** **Providing students with key and supporting concepts in software architecture.** **Assigning students to design software architectures.** **Assessment Methods:** **Homework assignments.** **Surprise quizzes.** **Monthly exams.** **Monthly reports.** **Final exams.** **Seminars.** **Affective and Value-Based Objectives:** **C1- Paying attention during lectures.** **C2- Participating in lectures.** **C3- Accepting the lecture content.** **C4- Having the desire to continue the lecture.**

10. **Course Structure**

Evaluation method	Learning method	Unit or subject name	Required Learning Outcomes	Hours	Week
Exam	Lecture	Introduction to Software Architecture	Understand the critical role of a software architect in modern .development	6	2-1
	Lecture	Principles of Software Architecture	Explore concepts like Separation of Concerns, Encapsulation, Modularity, Reusability, and Maintainability .	6	4-3
	Lecture	Software Architecture Models	Delve into Monolithic, Distributed Architectures, Microservices, and Event-Driven .Architectures	3	5
	Lecture	Design Patterns	Learn about MVC, MVP, MVVM, and essential backend patterns for effective .development	6	7-6
	Lecture	Architectural Documentation	Discover the importance of documentation, including	6	9-8

			Viewpoints, Perspectives, and UML .usage		
Lecture		Quality Attributes and Trade-offs	Explore quality attributes like performance, scalability, security, and .compliance	6	-10 11
Lecture		Architectural Styles	Understand different styles like Client-Server, Peer-to-Peer, Component-Based, and Publish-.Subscribe	3	12
Lecture		SOA and Microservices	Compare SOA with Microservices and understand nuances of Cloud-Native applications and Serverless .approaches	3	13
Lecture		Agile and Architecture	Integrate Agile practices with software architecture, covering Evolutive Design and Emergent .Architecture	3	14

		Lecture	Analysis and Strategy	Dive deep into requirements analysis, application type definition, data considerations, and testing .strategies	3	15	

11. Course Evaluation

Exams

12. Teaching and learning resources

Fundamentals of Software Architecture: A Practical Guide Kindle Edition by Anderson Rogério

Course Description Form

Course Name
Advanced Algorithms Design
Course Code
Semester / Year
2024-2025
The date of preparation this description
5/4/2025
Available Attendance Forms
Classrooms
Number of credit hours (total) / number of units (total)
2/2 ....
The name of the course administrator (if more than one name is mentioned)
Name: Assist.Prof.Dr. Sinan Diwan **Email: sdiwan@uowasit.edu.iq**
8. Course Objectives

Course Objectives	• **Understanding advanced concepts in algorithm design** • **Analyzing and designing advanced algorithms** • **Applying advanced algorithms to real-world problems** • **Evaluating and comparing algorithms** • **Understanding the theoretical foundations of advanced algorithms** • **Developing critical thinking and problem-solving**

	skills • Effective communication about algorithm design and analysis • Preparing for advanced research in the field of algorithms

9. Teaching and Learning Strategies

Strategy	• Interactive Lectures • Hands-On Lab Sessions • Flipped Classroom • Collaborative Group Work • Peer Review and Feedback • Case Studies and Real-World Examples • Scaffolded Assignments • Use of Digital Tools and Online Platforms • Continuous Assessment and Formative Feedback • Guest Lectures and Workshops

10. Course Structure

Evaluation method	Learning method	Unit or subject name	Required Learning Outcomes	Hours	Week
		Important of Algorithms		2	1
		Bubble Sort Algorithm		2	2
		Selection Sort Algorithm		2	3
		Quick Sort Algorithm		2	4
		Two Dimension sorting Algorithms		2	5
		Search Algorithms Strategies		2	6
		Sequential search Algorithm		2	7
		Binary search Algorithm		2	8
		Two Dimension search Algorithm		2	9
		Recursion Algorithms		2	10
		Recursion Functions Techniques		2	11

			Recursion Functions Techniques		2	12
			Recursion Functions Techniques		2	13
			Smart programming Algorithms		2	14
			Smart Agents Algorithms		2	15

11. Course Evaluation

12. Teaching and learning resources

Research Design:
Authors:

Course Description Form

Course Name
Networks applications
Course Code
Semester / Year
2/ 2024-2025
The date of preparation this description
Available Attendance Forms
Classrooms
Number of credit hours (total) / number of units (total)
....
The name of the course administrator (if more than one name is mentioned)
Name: Dr. Ali Fahem Alyasree   **Email:**
8. Course Objectives

Course Objectives	**A deep understanding of cloud computing, fog computing, and the Internet of Things (IoT).** **Analyze and design applications based on these technologies.** **Develop network application programming skills using modern technologies.**

	Address security issues and improve performance in these environments.

9. Teaching and Learning Strategies

Strategy	

10. Course Structure

Evaluation method	Learning method	Unit or subject name	Required Learning Outcomes	Hours	Week
		Introduction to networks, Cloud Computing, Fog Computing, and the Internet of Things			
		Cloud Computing Architecture			
		Cloud Computing: Platforms and Technologies			
		Cloud Computing Security			
		Fog Computing: Basic Concepts			
		Fog Computing Architecture			
		Fog Computing: Performance and Security Issues			
		Internet of Things (IoT): Introduction			
		IoT Application Development			
		IoT Data Management			
		IoT Security			

			Integrating cloud and fog computing with the IoT			
			Application Case Studies			
			Application Projects and Case Study Presentations			
			Comprehensive Review and Final Exam			

11. Course Evaluation

12. Teaching and learning resources

INTERNET OF THINGS Architecture and Design Principles
Raj Kamal
Fog and Edge Computing
Principles and Paradigms
Edited by Rajkumar Buyya and Satish Narayana Srirama
Cloud Computing
Sandeep Bhowmik

Course Description Form

Course Name
Advanced Artificial Intelligence
Course Code
Semester / Year
2/ 2024-2025
The date of preparation this description
Available Attendance Forms
Classrooms
Number of credit hours (total) / number of units (total)
3 / 3/ ....
The name of the course administrator (if more than one name is mentioned)
Name: Prof. Dr. Dheyaa Shaheed Sabr Al-Azzawi **Email:** ***dalazzawi@uowasit.edu.iq***
8. Course Objectives

Course Objectives	**1- Expand AI Knowledge – Introduce advanced concepts and methods in artificial intelligence with a focus on modern solutions.** **2- Analyze and Develop Intelligent Models – Enable students to analyze and develop advanced AI models using machine learning and deep learning techniques.** **3- Apply AI in Various Fields – Explore AI applications in computer vision, natural language**

	processing, robotics, healthcare, and business. 4- Handle Big Data – Develop skills to manage big data and utilize efficient algorithms for processing it. 5- Performance Analysis and Optimization – Study methods for evaluating AI model performance and improving it through optimization strategies. 6- Innovation and Research in AI – Encourage critical thinking and research in AI through projects and research reports

9. Teaching and Learning Strategies

Strategy	1- Interactive Lectures – Deliver theoretical concepts through interactive lectures with practical examples and real-world applications. 2- Project-Based Learning – Assign students practical projects to develop AI models and solve problems using modern techniques. 3- Case Study Analysis – Study and analyze real-world cases to illustrate AI applications in various fields. 4- Problem-Based Learning – Provide students with real-life problems to solve using appropriate algorithms and techniques. 5- Hands-On Labs – Conduct practical sessions to train students on tools and programming languages such as Python, TensorFlow, and PyTorch. 6- Discussions and Seminars – Encourage students to participate in discussions on AI advancements through presentations and scientific seminars. 7- Collaborative Learning – Promote teamwork in projects and experiments to enhance knowledge sharing and cooperation. 8- Continuous Assessment and Feedback – Use quizzes, assignments, and ongoing feedback to ensure students' understanding. 9- Utilizing Digital Resources – Leverage research papers, educational videos, and interactive platforms for self-learning. 10- Encouraging Scientific Research – Motivate students to read, review, and report on the latest innovations in AI.

10. Course Structure

Evaluation method	Learning method	Unit or subject name	Required Learning Outcomes	Hours	Week
1- Quiz	Interactive Lectures -2 Classroom -3	Introduction and history	□ Understand the concept of artificial intelligence and its	3	1

	Discussions		various definitions. □ Identify key historical developments in artificial intelligence. □ Recognize the relationship between AI and other fields such as computer science, mathematics, and neuroscience. □ Analyze the trends and challenges AI has faced over time.		
Class Participation Quiz	1- Interactive Lectures 2- Hands-on Exercises 3- Group Discussions	Predicate Calculus	□ Understand the concept of Predicate Calculus and its significance in AI. □ Differentiate between Propositional Logic and Predicate Logic. □ Utilize quantifiers (∀ for universal, ∃ for existential) to represent knowledge. □ Apply logical inference rules to solve problems using Predicate Calculus. □ Convert natural language statements into formal logical expressions	3	2
==	==	STRUCTURES AND STRATEGIES FOR STATE SPACE SEARCH	□ nderstand the concept of state space and its importance in AI. □ Learn how to represent problems using state space. □ Analyze different search strategies, such as Depth-First Search and Breadth-First Search. □ Apply uninformed and informed search algorithms to problem-solving. □ Compare the efficiency of various search strategies in terms of performance and time complexity. □ Utilize state space representation for real-world applications like AI games and automated planning	6	3-4
==	==	Heuristic Search	□ understand the concept of heuristic	6	5-6

			search and its importance in AI. □ Differentiate between uninformed search and heuristic search strategies. □ Analyze and utilize evaluation functions and heuristics to improve search efficiency. □ Implement heuristic search algorithms such as A*, Greedy Best-First Search, and Hill Climbing. □ Compare heuristic search algorithms based on efficiency and time complexity. □ Apply heuristic search to AI applications such as planning, games, and robotics		
		Test		3	7
==	==	BUILDING CONTROL ALGORITHMS FOR STATE SPACE SEARCH	□ Understand the role of control algorithms in state space search in AI. □ Learn the principles of constructing search algorithms in state space. □ Analyze and develop search algorithms like Depth-First Search (DFS), Breadth-First Search (BFS), and A*. □ Design and implement advanced search control algorithms such as Iterative Deepening Search and Bidirectional Search. □ Enhance search performance using techniques like heuristics and pruning. □ Apply search control algorithms to real-world problems like planning, games, and robotics	6	8 – 9

	==	==	Knowledge Representation	• Understand the concept of knowledge representation in AI and its objectives. • Identify different methods for knowledge representation, such as Semantic Networks, Frames, and Logic. • Distinguish between implicit and explicit knowledge representation. • Develop and apply techniques for knowledge representation in AI, such as Neural Networks and Symbolic Logic. • Evaluate knowledge representation strategies based on efficiency, flexibility, and understandability. • Apply knowledge representation to solve AI problems like planning and machine learning.	6	11 - 10
	==	==	MACHINE LEARNING: SYMBOL-BASED	Understand the theoretical foundations of symbol-based machine learning. • Identify different types of symbolic machine learning algorithms, such as	6	13 - 12

				Supervised Learning, Unsupervised Learning, and Reinforcement Learning. • Understand how symbolic algorithms can be applied to AI problems such as classification, clustering, and knowledge planning. • Analyze techniques like Symbolic Decision Making and Expert Systems, and their use in machine learning. • Apply symbolic machine learning algorithms using techniques such as symbolic reasoning. • Compare symbolic and non-symbolic machine learning approaches in terms of efficiency and accuracy.		
==	==	MACHINE LEARNING: CONNECTIONIST	• Understand the theoretical foundations of connectionist machine learning (neural networks). • Identify different types of neural networks, such as Multilayer Perceptrons (MLP), Convolutional Neural	3	14	

				Networks (CNN), and Recurrent Neural Networks (RNN). • Understand how neural networks are applied to tasks such as classification, pattern recognition, and predictive systems. • Learn how to train neural networks using techniques like Backpropagation and optimization algorithms like Gradient Descent. • Analyze the applications of connectionist learning in fields such as computer vision, natural language processing, and predictive learning. • Evaluate neural network performance using metrics such as accuracy and mean squared error (MSE).		
			Test		3	15

11. Course Evaluation

12. Teaching and learning resources

Artificial Intelligence Structures and strategies for complex Problem Solving, George Luger, 2023, 6th edition.