هيئة تقويم التعليم والتدريب
Education & Training Evaluation Commission

T-104
2022

Course Specification

هيئة تقويم التعليم والتدريب
Education & Training Evaluation Commission

T-104
2022

Course Specification

Course Title:	**Web Programming**
Course Code:	**CS 210**
Program:	**Computer Science (Programming)**
Department:	**Applied Sciences**
College:	**Applied Collage**
Institution:	**Al Imam Muhammad bin Saud Islamic University**
Version:	**1**
Last Revision Date:	**20/06/2023**

Table of Contents:

A. General information about the course:

Course Identification					
1. Credit hours:	3 hours (2 hours Lecture, 2 hours Lab)				
2. Course type:					
a.	University ☐	College ☐	Department☐	Track☒	Others☐
b.	Required ☒	Elective☐			
3. Level/year at which this course is offered: Fourth/ Two					
4. Course general Description: This course aims to equip students with the knowledge and the skills necessary to create dynamic websites and web applications. Students will learn how to design and develop interactive web pages, create, and manage databases, and deploy web applications to production servers. Throughout the course, students will work on a variety of projects that will challenge them to apply their knowledge and skills in practical ways. By the end of the course, students will have a solid foundation in web programming and be well-equipped to pursue further study or a career in the field.					
5. Pre-requirements for this course (if any): CS119					
6. Co- requirements for this course (if any): None					
7. Course Main Objective(s) This course aims to enable the students to build a dynamic website.					

1. Teaching mode (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1.	Traditional classroom		
2.	E-learning		
3.	Hybrid • Traditional classroom • E-learning	44	100%
4.	Distance learning		

2. Contact Hours (based on the academic semester)

No	Activity	Contact Hours
1.	Lectures	22
2.	Laboratory/Studio	22
3.	Field	
4.	Tutorial	
5.	Others (specify)	110
	Total	154

B. Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Code of CLOs aligned with program	Teaching Strategies	Assessment Methods
1.0	Knowledge and understanding			
1.1	Knowledge of the essential and general concepts of JavaScript, and the differences between JavaScript and other programming languages.	ع1، ع2، ع4، ع5	-Classroom Lectures. -Discussions. -Survey. -Experimental Learning. -Self Learning. -Development Lectures. -Brainstorming - Web Survey. -KWL - Learning Schedule. - Mind Maps. -Concept Maps.	-In-person and Online Exams. -Classroom Questions. -Assignments. -Participations. - Presentations -Discussions. -Debates. -Cognitive Tests. -Student Activity File.
2.0	Skills			
2.1	The ability to use the basic commands and statements in JavaScript effectively, and to validate user inputs.	م1، م2، م3، م4، م5، م7	-Labs. -Development Lectures. -Experimental Learning. -Peers Learning. -Self Learning. -Discussions. -Participations. - Web Survey. -Brainstorming. -Teamwork. -Problem Solving. -Projects. -Electronic Discussion Groups	- Presentations - Rubrics. - Auditions. -Productivity Measurements. -Observations. -Assignments. -Programming Projects. -Student Activity File. -Peers Assessments. -Self Assessments.
2.2	Using the control statements effectively.	م1، م2، م3، م4، م5، م7		
2.3	The ability to utilize functions and methods in JavaScript.	م1، م2، م3، م4، م5، م7		
2.4	The ability to use the JavaScript standard libraries.	م1، م2، م3، م4، م5، م7		
2.5	The process of constructing websites using software that incorporates the specific features of the programming languages utilized in the course.	م1، م2، م3، م4، م7		
2.6	The use of information and communication technology in communication, exchanging ideas, scientific research, and tasks accomplishments.	م1، م2، م7		
2.7	Practicing critical thinking and solving problems that	م1، م2، م7		

Code	Course Learning Outcomes	Code of CLOs aligned with program	Teaching Strategies	Assessment Methods
	the learner faces in the course in creative ways.			
3.0	Values, autonomy, and responsibility			
3.1	Cooperation, teamwork, and professional ethics.	ق1	-Projects. -Teamwork. -Discussions. - Participations. -Labs. -Role Modeling. - Web Survey.	-Notes. -Discussions. -Participations. -Classroom Questions. -Rubrics. -Scale Measurements. -Peers Assessments. -Self Assessment. -Student Activity File.
3.2	Take responsibility for continuous learning and continuing personal development.	ق2		
3.3	Efficient and effective time management when applying acquired knowledge and skills.	ق3		

C. Course Content

No	List of Topics	Contact Hours
1.	Introduction to JavaScript: • A Review of the most important commands in HTML and CSS languages. • Overview of JavaScript. • Advantages of JavaScript. • The differences between HTML, CSS, and JavaScript. • The difference between Java and JavaScript. • JavaScript Prosperities.	2 hours
2.	Fundamentals of JavaScript: • Simple programming in JavaScript: ○ Learning the basic syntax and structure in JavaScript. ○ How to incorporate JavaScript into a page. ▪ Adding the JavaScript to a page's HTML. ▪ Using external JavaScript file. • Input and output of data in JavaScript. • Components of the JavaScript language: ○ Comments. ○ Literals. ○ Numbers. ○ Variables. ▪ Declaration. ▪ Initialization.	4 hours

	○ Constants. ○ Reserved Words. ○ Operators: ▪ Assignment operators. ▪ Arithmetic operators. ▪ Comparison operators. ▪ Relational operators.	
3.	Control statements: • Understanding the concept of control statements: ○ Condition statements: ▪ Conditional sentences tools: (structural form, explanation of general form, how it works, examples). • Simple if statement. • if … else statement. • Nested if … else statement. • switch statement. ▪ Common mistakes in conditional sentences. ○ Looping structures: ▪ Repetition sentences tools: (structural form, explanation of general form, how it works, examples). • For statement. • While statement. • Do While statement. ▪ Common errors in looping structures. ○ Jump statements: ▪ Jump sentences tools: (how it works, where it should be used, examples). • Break statement • Continue statement. • Return statement. ▪ Comparison between jumping statements.	8 Hours
4.	Functions: • The concept of functions. • The purpose of using functions. • Defining and invoking functions. • Function types: ○ Built-in functions in JavaScript ▪ Input functions. ▪ Output functions. ▪ Operations functions. ▪ Practical examples of all functions built into JavaScript. ○ Functions created by JavaScript programmers: • Function arguments and parameters. • Function calls:	8 Hours

	○ Call the function from within the program. ○ Call the function from outside the program. • Returning values from functions. • Variables scope: ○ Local variables. ○ Global variables. • Function operators. • Common mistakes when writing a function.	
5.	Events: • Understanding Events. • Definition, usage, and practical examples. • Event types: ○ Mouse events: ▪ onclick event. ▪ ondblclick event. ▪ onmouseenter event. ▪ onmouseleave event. ▪ onmousemove event. ▪ onmousout event. ○ keyboard events: ▪ onkeyup event. ▪ onkeydown event. ▪ onkeypress event. ○ Form events: ▪ onsubmit event. ▪ onreset event. ▪ onclick event. ▪ onchange event. ▪ onblur event. ○ Page events: ▪ Onload event. ▪ Onunload event.	6 Hours
6.	Validation of entered data and use of cookies: • Verify that no field is left blank (Required Field). • Control the length of the data entered in the field. • Matching data entered in two fields. • Prevent entering any data type other than the field type. • Examples to verify the validity of the entered data. • Using cookies.	2 Hours
7.	Standard libraries in JavaScript: • Objects ○ Creating objects. ○ Adding properties to an object. • Arrays: ○ What is an Array? ○ Array types: ▪ one-dimensional array.	10 Hours

	▪ multidimensional array. ○ Creating Arrays. ○ Array representation. ○ Reading and writing Array elements: ▪ Assignment with the for loop ▪ Assignment with the for in loop ▪ Assignment by mentioning the name of the array only. ○ Array Methods: (function classification, function, method of calling it, Application examples ▪ Join. ▪ Concat. ▪ Pop. ▪ Push. ▪ Reverse. ▪ Sort. ▪ Indexof. ▪ Includes. ▪ Length. • Math Objects: ○ Most important methods. ○ Purpose of each method. ○ Examples. • Date Objects: ○ Most important methods. ○ Purpose of each method. ○ Examples. ○ Types of Date object functions: ▪ Functions to return values from date. ▪ Functions to put values in the date. ▪ Functions to convert the date. • String Objects: ○ Most important string and conversion methods. ○ Purpose of each method. ○ Examples. • Window Objects: ○ Most important methods. ○ Purpose of each method. ○ Examples. ○ Windows Objects. ○ Document Objects. ○ History Objects. ○ Location Objects.	
8.	Building websites using application software and applying the characteristics of the languages used in the course.	4 Hours
Total		44

D. Students Assessment Activities

No	Assessment Activities *	Assessment timing (in week no)	Percentage of Total Assessment Score
1.	One Midterm Exam	Week 6	15
2.	Quizzes	The whole semester	10
4.	Labs	The whole semester	15
5.	Project	Week 10	10
6.	Attendance and Classroom Participation	The whole semesters	5
7.	Final Exam	Week 12	40
8.	Total		100

*Assessment Activities (i.e., Written test, oral test, oral presentation, group project, essay, etc.)

E. Learning Resources and Facilities

1. References and Learning Resources

Essential References	JavaScript The Definitive Guide, By David Flanagan, 6th Edition.
Supportive References	PHP, MySQL, JavaScript, and HTML5 All in One for Dummies, By Steven Suehring and Janet Valade, 1st Edition. JavaScript Bible, By Danny Goodman, Gold Edition.
Electronic Materials	Course Lectures on the Blackboard.
Other Learning Materials	N/A

2. Required Facilities and equipment

Items	Resources
facilities (Classrooms, laboratories, exhibition rooms, simulation rooms, etc.)	Lecture room with Smart board Lab with 25 PCs.
Technology equipment (Projector, smart board, software)	PC and WIFI Internet access within the classroom Projector Visual Studio Code او Notepad++
Other equipment (Depending on the nature of the specialty)	A software for building websites determined by course faculty.

F. Assessment of Course Quality

Assessment Areas/Issues	Assessor	Assessment Methods
Effectiveness of teaching strategies	Students – Peers	1. Questionnaires and surveys approved by the department. 2. Peers' assessments by faculties. 3. Reviewing the results of the students' evaluation.
Effectiveness of students' assessment	Peers, Program Leaders, Faculty, Students.	1. Questionnaires and surveys approved by the department. 2. Review course specifications and course reports periodically. 3. Peers' assessments. 4. Review samples of students works.
Learning Resources	Program Leaders, Faculty, Students.	1. Questionnaires and surveys approved by the department. 2. Students transcripts.
Quality of learning resources	Program Leaders, Faculty.	1. Review course report. 2. Analyze exam models and students grade records.
The extent to which CLOs have been achieved		
Other		

Assessor (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify)
Assessment Methods (Direct, Indirect)

G. Specification Approval Data

COUNCIL /COMMITTEE	COMPUTER PROGRAMS DEVELOPMENT COMMITTEE
REFERENCE NO.	FIRST- THE THIRD SEMESTER OF THE YEAR 1444
DATE	20 / 06 / 2023 G, 02/ 12 /1444 H

هيئة تقويم التعليم والتدريب
Education & Training Evaluation Commission

T-240
2023

Course Specification

هيئة تقويم التعليم والتدريب
Education & Training Evaluation Commission

T-240
2023

Course Specification

Course Title:	**Operating Systems Management** (OSM)
Course Code:	CS241
Program:	Programming Technology
Department:	Applied Sciences
College:	Applied College
Institution:	Imam Mohammad Ibn Saud Islamic University
Version:	2023
Last Revision Date:	August 1st, 2023

Table of Contents:

A. General information about the course:

Course Identification					
1. Credit hours:	2 (1 theory, 2 lab)				
2. Course type					
a.	University ☐	College ☒	Department☐	Track☐	Others☐

b. Required ☒ Elective☐	
3. Level/year at which this course is offered:	Third
4. Course general Description: ○ Operating systems are an essential part of any computer system. Similarly, a course on operating systems is an essential part of any computer science education. This field is undergoing rapid change, as computers are now prevalent in virtually every arena of day-to-day life—from embedded devices in automobiles through the most sophisticated planning tools for governments and multinational firms. Yet the fundamental concepts remain fairly clear, and it is on these that we base this course.	
5. Pre-requirements for this course (if any): CS101	
6. Co- requirements for this course (if any): None	
7. Course Main Objective(s): ○ The main objective of operating systems is to make computers and other devices easier for us to use. The OS can deliver greater convenience, productivity, and ease of use by providing a user interface that allows us to interact with our hardware and software. It simplifies tasks by making previously complex processes automatic, giving us less to worry about when using our computers.	

1. Teaching mode (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1.	Traditional classroom		
2.	E-learning		
3.	Hybrid • Traditional classroom • E-learning	3 hours\week	100%
4.	Distance learning		

2. Contact Hours (based on the academic semester)

No	Activity	Contact Hours
1.	Lectures	18
2.	Laboratory/Studio	18
3.	Field	
4.	Tutorial	
5.	Others (Specify)	
	Total	36

B. Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Code of CLOs aligned with program	Teaching Strategies	Assessment Methods
1.0	Knowledge and understanding			
1.1		٤1، ٤5		

Code	Course Learning Outcomes	Code of CLOs aligned with program	Teaching Strategies	Assessment Methods
	Familiarity with the basic and general concepts of operating systems of all kinds.		- Lecture. - Discussion. - Survey. - Discovery learning. - Self-education. - Developed lecture. - Brainstorming. - Web survey. - KWL chart. - Mind maps. - Concept maps.	- Traditional and online achievement tests. - Questions. - Assignments and assessments. - Presentations. - Discussion and debates. - Cognitive performance tests. - Achievement file.
1.2	Knowledge of the basic principles of operations management in various operating systems.	ع1، ع5		
1.3	Understand how operating systems work and manage them.	ع1، ع5		
2.0	Skills			
2.1	Ability to work with the operating system (Windows).	م1، م2، م7	- Demonstration. - Developed lecture. - Discovery learning. - Peer learning. - Self-education. - Discussion. - Web survey. - Brainstorming. - Co-learning. - Problem Solving. - Project. - Online discussion.	- Presentations. - Rating ladders. - Performance tests. - Production metrics. - Observation. - Projects. - Achievement file. - Peer assessment. - Self-calendar.
2.2	Proficiency in using the operating system (Dos).	م1، م2، م7		
2.3	The ability to deal with network operating systems.	م1، م2، م7		
2.4	Using information and communication technology in communication, exchanging ideas, scientific research, and performing tasks and costs.	م1، م2، م7		
2.5	Practicing critical thinking and solving problems that the learner faces in the course in creative ways.	م1، م2، م7		
3.0	Values, autonomy, and responsibility			
3.1	Cooperation, teamwork, and professional ethics.	ق1	- Demonstration. - Developed lecture. - Discovery learning. - Peer learning. - Self-education. - Discussion. - Web survey. - Brainstorming. - Co-learning. - Problem Solving.	- Presentations. - Rating ladders. - Performance tests. - Production metrics. - Observation. - Projects. - Achievement file. - Peer assessment. - Self-calendar.
3.2	Take responsibility for continuous learning and continuing personal development.	ق٢		
3.3		ق٣		

Code	Course Learning Outcomes	Code of CLOs aligned with program	Teaching Strategies	Assessment Methods
	Efficient and effective time management when applying acquired knowledge and skills.		- Project. - Online discussion.	

C. Course Content

No	List of Topics	Contact Hours
1.	**Introduction to Operating System Management** • Definition and functions of an operating system. • Evolution of operating systems. • Types of operating systems (e.g., Windows, Linux, macOS).	3
2.	**Process Management** • Process concept and process control block. • Process scheduling algorithms. • Inter-process communication and synchronization. • Deadlock detection and prevention.	3
3.	**Memory Management** • Memory hierarchy and virtual memory. • Memory allocation strategies (e.g., paging, segmentation). • Memory management unit (MMU) and address translation. • Memory fragmentation and compaction.	3
4.	**File System Management** • File system organization and structure. • File operations (e.g., create, open, read, write, delete). • File allocation methods (e.g., contiguous, linked, indexed). • File system security and access control.	3
5.	**Device Management** • I/O devices and drivers. • I/O buffering and spooling. • Disk scheduling algorithms. • RAID (Redundant Array of Independent Disks) and its levels.	3
6.	**Process Synchronization and Communication** • Critical sections and mutual exclusion.	3

	• Semaphores and monitors. • Message passing and shared memory. • Readers-writers problem and dining philosophers' problem.	
7.	**Process and Thread Management** • Forking and process spawning. • Thread creation and synchronization. • Thread scheduling and CPU utilization. • Multithreading advantages and challenges.	3
8.	**Security and Protection** • Security threats and vulnerabilities. • Access control models (e.g., discretionary, mandatory). • User authentication and authorization. • Security mechanisms and policies.	3
9.	**Operating System Performance Evaluation and Optimization** • Performance metrics and benchmarks. • Performance monitoring and profiling tools. • Bottleneck identification and optimization techniques. • CPU scheduling, disk I/O, and memory optimization strategies.	6
10.	**Case Studies and Emerging Trends** • Study of real-world operating systems (e.g., Windows 10, Ubuntu, macOS Catalina). • Cloud-based operating systems (e.g., Google Chrome OS). • Virtualization and containerization technologies (e.g., VMware, Docker). • Internet of Things (IoT) and operating system requirements.	6
Total		36

D. Students Assessment Activities

No	Assessment Activities *	Assessment timing (In week no)	Percentage of Total Assessment Score
1.	Midterm Exam	Week7	20%
2.	Quizzes	Continuous	10%
3.	Homework	Continuous	8%
4.	Participation	Continuous	2%

No	Assessment Activities *	Assessment timing (In week no)	Percentage of Total Assessment Score
5.	Practical Assessment	Continuous	20%
6.	Final Exam	Week13	40%

*Assessment Activities (i.e., Written test, oral test, oral presentation, group project, essay, etc.)

E. Learning Resources and Facilities

1. References and Learning Resources

Essential References	• Operating system concepts by Silberschatz Galvin.
Supportive References	○ Modern Operating Systems by Andrew Tanenbaum.
Electronic Materials	○ Online resources will be provided during class lectures.
Other Learning Materials	N/A

2. Required Facilities and equipment

Items	Resources
Facilities (Classrooms, laboratories, exhibition rooms, simulation rooms, etc.)	Classroom included: 1. A projector connected to a PC, preferably with Internet access. 2. A vertical sliding board. 3. An equipped computer lab with at least 25 seats.
Technology equipment (Projector, smart board, software, etc.)	Computing resources (Projector, data show, Smart Board, software: Win Dos- Win Server- Linux - Ubuntu, etc.).
Other equipment (Depending on the nature of the specialty)	N/A

F. Assessment of Course Quality

Assessment Areas/Issues	Assessor	Assessment Methods
Effectiveness of teaching	Student	1. Students feedback (collected through surveys) as per university policy/procedure. 2. Teacher's Course report.

Assessment Areas/Issues	Assessor	Assessment Methods
Effectiveness of student's assessment	Faculty	1. Review of Course Reports. 2. Review of Student feedback.
Quality of learning resources	Student and Faculty	Indirect using course evaluation and faculty survey.
The extent to which CLOs have been achieved	Program Leaders	Continuous review of the course contents, teaching strategies and utilizing the best practices.
Other	N/A	N/A

Assessor (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify)
Assessment Methods (Direct, Indirect)

G. Specification Approval Data

COUNCIL /COMMITTEE	
REFERENCE NO.	
DATE	

هيئة تقويم التعليم والتدريب
Education & Training Evaluation Commission

اعتماد NCAAA T4 2020

Course Specifications

Course Title:	Programming 2
Course Code:	CS 220
Program:	Computer Science (Programming)
Department:	Applied Sciences
College:	Applied College
Institution:	Imam Muhammad Bin Saud Islamic University

Table of Contents

A. Course Identification

1. Credit hours: 3(2 theory , 2 lab)
2. Course type **a.** University ☐ College ☐ Department ☒ Others ☐ **b.** Required ☒ Elective ☐
3. Level/year at which this course is offered: 4th Semester
4. Pre-requisites for this course (if any)**:** CS 120
5. Co-requisites for this course (if any)**:** None

6. Mode of Instruction (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	**Traditional classroom**		
2	**Blended**	44	100%
3	**E-learning**		
4	**Distance learning**		
5	**Other**		

7. Contact Hours (based on academic semester)

No	Activity	Contact Hours
1	**Lecture**	22
2	**Laboratory/Studio**	22
3	**Tutorial**	
4	**Others** (specify)	110
	Total	154

B. Course Objectives and Learning Outcomes

1. Course Description

The course covers the basics of object programming by addressing the subject of factions as a software structure in a unified format that is easy to use, as well as building and calling functions, and using dimensional and two-dimensional arrays as an example of a complex data architecture. Over the course of the semester, the course includes an integrated case study in which all the previous tools are employed as well as the experiences gained in the relevant courses to build an integrated project.

2. Course Main Objective

The course aims to give the student higher skills in programming science to be able to propose solutions to problems in new ways and formulate them in the form of a computer program, and the ability to write programs in a high-level programming language to solve this type of problems.

3. Course Learning Outcomes

	CLOs	Aligned PLOs
1	**Knowledge and Understanding**	
1.1	Familiarity with the basic and detailed concepts of object-oriented programming (OOP).	ع1، ع2، ع5
2	**Skills :**	
2.1	Build a complex data architecture using one and two-dimensional arrays.	م1، م2، م3، م4، م7
2.2	Constructing and calling functions with their various classifications.	م1، م2، م3، م4، م7
2.3	Programming problem solving using objects and factions.	م1، م2، م3، م4، م7
2.4	Mastering the characteristics of object-oriented programming.	م1، م2، م3، م4، م7
2.5	Mastering the mechanism of tracking the progress of the implementation of the program.	م1، م2، م7
2.6	Use of information and communication technology in communication, exchange of ideas, scientific research, performance of tasks and costs.	م1، م2، م7
2.7	Practice critical thinking and solving problems facing the learner in the course in creative ways.	م1، م2، م7
3	**Values:**	
3.1	Cooperation, teamwork and professional ethics.	ق1
3.2	Take responsibility for continuous learning and continued personal development.	ق2
3.3	Manage time efficiently and effectively when applying the acquired knowledge and skills.	ق3

C. Course Content

No	List of Topics	Contact Hours
1	**Arrays:** • Matrix concept. • Matrix definition. • Matrix types: o Single matrix. o Multidimensional matrix. • Identify the elements of the matrix. • Access the matrix. • Operations on a single matrix. • Operations on the binary matrix.	8
2	**Methods:** • The object of functions.	16

	• The general form of the function. • Classifications of functions: o Classification by type: ✓ Build-in. ✓ User-defined functions. o Classification in terms of accessibility: ✓ Public. ✓ Private. ✓ Protected. ✓ Default (Package private). o Classification in terms of participation: ✓ Shared functions (Static). ✓ Non-shared functions (None-static) (Instance). o Classification in terms of return value: ✓ Functions that return value. ✓ Functions do not return (Return no value). o Classification in terms of transactions: ✓ Functions that need parameters (Have parameter). ✓ Functions that do not need parameters (Have no parameter). • Flow of execution. • Build and call functions with their varieties in the program. • Method Overloading.	
3	**Classis & Objects:** • Class Declaration • Class properties and Class Methods • Object Creation • Accessing properties and method • Constructors •Recall the constructors and how to use them.	16
4	**A set of object-oriented programming characteristics:** • Modifiers: variables, functions, objects. • Scope. • life time.	4
Total		44

D. Teaching and Assessment

1. Alignment of Course Learning Outcomes with Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
1.0	**Knowledge and Understanding**		
1.1	Familiarity with the basic and detailed concepts of object-oriented programming (OOP).	- Classroom lecture - Dialogue and discussion -Survey - Discovery learning - Self-learning	- Traditional and electronic achievement tests. - Classroom questions

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
		- Developed lecture -Brainstorming - Web Polling - KWL Learning Schedule - Mind maps - Concept maps	- Assignments and periodic evaluations - Presentations - Discussion and debate - Cognitive performance tests - Achievement file
2.0	**Skills**		
2.1	Build a complex data architecture using one and two-dimensional arrays.	- Practical presentation - Developed lecture - Discovery learning - Peer learning - Self-learning - Dialogue and discussion - Web Polling -Brainstorming - Cooperative learning - Problem solving - Project-based learning - E-discussion forums.	- Presentations - Grading scales - Performance tests - Production metrics - Observation - Software projects - Achievement file - Peer Evaluation - Self-evaluation
2.2	Constructing and calling functions with their various classifications.		
2.3	Programming problem solving using objects and factions.		
2.4	Mastering the characteristics of object-oriented programming.		
2.5	Mastering the mechanism of tracking the progress of the implementation of the program.		
2.6	Use of information and communication technology in communication, exchange of ideas, scientific research, performance of tasks and costs.		
2.7	Practice critical thinking and solving problems facing the learner in the course in creative ways.		
3.0	**Values**		
3.1	Cooperation, teamwork and professional ethics.	-Project-based learning - Cooperative learning - Dialogue and discussion - Practical lecture - Modeling and role models - Cross-polling Web	- Note cards - Discussion and dialogue - Classroom questions - Grading scales - Measures of values - Self-evaluation - Peer Evaluation - Achievement file
3.2	Take responsibility for continuous learning and continued personal development.		
3.3	Manage time efficiently and effectively when applying the acquired knowledge and skills.		

2. Assessment Tasks for Students

#	Assessment task*	Week Due	Percentage of Total Assessment Score
1	Midterm	Week 6	15 %
2	Quizzes (3-4 Quizzes)	All Semester	10 %
3	Lab Evaluation	All Semester	20 %
4	Project	Week 10	10 %
5	Participation	All Semester	5 %

#	Assessment task*	Week Due	Percentage of Total Assessment Score
6	Final	Week 13	40 %
7	**Total**		**100 %**

***Assessment task** (i.e., written test, oral test, oral presentation, group project, essay, etc.)

E. Student Academic Counseling and Support

Arrangements for availability of faculty and teaching staff for individual student consultations and academic advice :

- Distribution of guidelines prepared by the Deanship of Admission and Registration Affairs.

● Allocate office hours for faculty members to follow up on students' academic inquiries and communicate via mail

Electronic and course page on the Blackboard system, activating discussions, support and providing feedback.

● Studying the academic problems of college students and everything related to the reasons for dismissal, academic stumbling, delay and low average.

F. Learning Resources and Facilities

1.Learning Resources

Required Textbooks	Java: how to program, by P.J. Deitel and H.M. Deitel
Essential References Materials	1. Head First Java, by Kathy Sierra and Bert Bates. 2. Java: A Beginner's Guide, by Herbert Schildt. 3. Effective Java: Programming Language Guide (Java Series), by Joshua Bloch. 4. Simple Program Design, by Lesley Robertson.
Electronic Materials	The course website on the Internet through the Blackboard platform, through which assignments are submitted electronically, discussion questions are asked, and the course content is uploaded so that the student can return to lectures in full, publish electronic quizzes, and educational videos related to the course, and advertisements for the course are displayed and direct feedback is provided according to what the course professor sees.
Other Learning Materials	-

2. Facilities Required

Item	Resources
Accommodation (Classrooms, laboratories, demonstration rooms/labs, etc.)	- Classroom - Computer lab.
Technology Resources (AV, data show, Smart Board, software, etc.)	- Smart board - Data projector -NetBeans.
Other Resources (Specify, e.g. if specific laboratory equipment is required, list requirements or attach a list)	-

G. Course Quality Evaluation

Evaluation Areas/Issues	Evaluators	Evaluation Methods
The effectiveness of teaching methods.	Peer References – Students	1. Questionnaires and referendums approved by the department. 2. Peer evaluation of faculty members. 3. Review the results of students' evaluation.
The effectiveness of student assessment methods.	Peer references – program leaders – faculty members – students	1. Questionnaires and referendums approved by the department. 2. Review course descriptions and course reports periodically. 3. Peer evaluation and periodic exchange of correction and scrutiny between faculty colleagues. 4. Review samples of students' work.
Learning Resources.	Program Leaders – Faculty Members – Students	1. Questionnaires and referendums approved by the department. 2. Write-off and monitoring lists.
The extent to which the learning outcomes of the course have been achieved.	Program Leaders – Faculty Members	1. Review the course report. 2. Analyze test forms, grades, student work and achievement records

Evaluation areas (e.g., Effectiveness of teaching and assessment, Extent of achievement of course learning outcomes, Quality of learning resources, etc.)
Evaluators (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify)
Assessment Methods (Direct, Indirect)

H. Specification Approval Data

Council / Committee	
Reference No.	
Date	

T-250
2023

Course Specification

T-250
2023

Course Specification

Course Title:	Systems Analysis and Design
Course Code:	CS0250
Program:	Programming Technology
Department:	Applied Sciences
College:	Applied College
Institution:	Imam Mohammad Ibn Saud Islamic University
Version:	2023
Last Revision Date:	August 1st, 2023

Table of Contents:

A. General information about the course:

Course Identification	
1. Credit hours:	3 (2 theory, 2 lab)

2. Course type	
a.	University ☐ College ☒ Department☐ Track☐ Others☐
b.	Required ☒ Elective☐
3. Level/year at which this course is offered:	Third

4. Course general Description:

- Systems Analysis and Design (SAD) is an exciting, active field in which analysts continually learn new techniques and approaches to develop systems more effectively and efficiently. However, there is a core set of skills that all analysts need to know, no matter what approach or methodology is used. All information systems projects move through the four phases of planning, analysis, design, and implementation; all projects require analysts to gather requirements, model the business needs, and create blueprints for how the system should be built; and all projects require an understanding of organizational behavior concepts like change management and team building.

- This course captures the dynamic aspects of the field by keeping students focused on doing SAD while presenting the core set of skills that we feel every systems analyst needs to know today and in the future.

5. Pre-requirements for this course (if any): CS120

6. Co- requirements for this course (if any): None

7. Course Main Objective(s):

- The goal of this course is to enable students to do SAD—not just read about it but understand the issues so that they can actually analyze and design systems. The course introduces each major technique, explains what it is, explains how to do it, presents an example, and provides opportunities for students to practice before they do it in a real-world project. After learning each chapter, the student will be able to perform that step in the system development life cycle (SDLC) process.

1. Teaching mode (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1.	Traditional classroom		
2.	E-learning		
3.	Hybrid • Traditional classroom • E-learning	4hours\week	100%
4.	Distance learning		

2. Contact Hours (based on the academic semester)

No	Activity	Contact Hours
1.	Lectures	24
2.	Laboratory/Studio	24
3.	Field	
4.	Tutorial	

5.	Others (Specify)	
	Total	48

B. Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Code of CLOs aligned with program	Teaching Strategies	Assessment Methods
1.0	Knowledge and understanding			
1.1	Familiarity with aspects of information systems analysis and design and common concepts between them.	ع1، ع5	- Lecture. - Discussion. - Survey. - Discovery learning. - Self-education. - Developed lecture. - Brainstorming. - Web survey. - KWL chart. - Mind maps. - Concept maps.	- Traditional and online achievement tests. - Questions. - Assignments and assessments. - Presentations. - Discussion and debates. - Cognitive performance tests. - Achievement file.
1.2	Distinguishing the development life cycle of software information systems with its stages, activities and tools used in it.	ع1، ع5		
1.3	Knowledge of the characteristics, advantages, and disadvantages of methodologies for developing software systems.	ع1، ع5		
2.0	Skills			
2.1	Differentiate between methodologies and methods for developing software systems.	م1، م2، م7	- Demonstration. - Developed lecture. - Discovery learning. - Peer learning. - Self-education. - Discussion. - Web survey.	- Presentations. - Rating ladders. - Performance tests. - Production metrics. - Observation. - Projects.
2.2	Apply planning tools to make decisions about the possibility of implementing the proposed system.	م1، م2، م7		
2.3	Analyze the main and sub-problem parts and solve them using the scientific method.	م1، م2، م3، م4، م5، م7		
2.4	Analyzing the needs of existing systems and converting them	م1، م2، م3، م4، م5، م7		

Code	Course Learning Outcomes	Code of CLOs aligned with program	Teaching Strategies	Assessment Methods
	into functional and non-functional requirements.		- Brainstorming. - Co-learning. - Problem Solving. - Project. - Online discussion.	- Achievement file. - Peer assessment. - Self-calendar.
2.5	Modeling the basic elements of systems using analysis and design tools, in preparation for starting the programming process with high quality.	م1، م2، م3، م4، م5، م6، م7		
2.6	Using information and communication technology in communication, exchanging ideas, scientific research, and performing tasks and costs.	م1، م2، م7		
2.7	Practicing critical thinking and solving problems that the learner faces in the course in creative ways.	م1، م2، م7		
3.0	Values, autonomy, and responsibility			
3.1	Cooperation, teamwork, and professional ethics.	ق1	- Demonstration. - Developed lecture. - Discovery learning. - Peer learning. - Self-education. - Discussion. - Web survey. - Brainstorming. - Co-learning. - Problem Solving. - Project. - Online discussion.	- Presentations. - Rating ladders. - Performance tests. - Production metrics. - Observation. - Projects. - Achievement file. - Peer assessment. - Self-calendar.
3.2	Take responsibility for continuous learning and continuing personal development.	ق٢		
3.3	Efficient and effective time management when applying acquired knowledge and skills.	ق٣		

C. Course Content

No	List of Topics	Contact Hours

1.	**Introduction to the analysis and design of information systems:** ▪ Introduction. • Advantages of the Information Revolution • Systems and Information. ○ System (definition - components - types). ○ Information (definition - information, data, and knowledge). • Information System. ○ What do we mean by information system? ○ Components of the information system. ○ Stakeholders. ○ Classifications of information systems. ○ The importance of information systems. ○ Benefits of information systems.	4
2.	**Software Development Lifecycle (SDLC)** ▪ System Development Lifecycle. • Phases of System Development Life Cycle: ○ Planning Phase. ○ Analysis Phase. ○ Design Phase. ○ Implementation Phase. ○ Testing Phase. ○ Maintenance Phase. • Cross Lifecycle Activities: ○ Project Management. ○ Requirements Management. ○ Quality Assurance Management. ○ Configuration & Control Management.	6
3.	**Software engineering (methods and methodologies)** • Systems development methods: ○ Linear. ○ Iterative. • System development methodologies: characteristics, advantages, disadvantages, appropriate and inappropriate cases for each of the following: ○ Waterfall Model. ○ Prototyping Model. ○ Incremental Model. ○ Spiral Model.	4
4.	**Planning stage tools:**	18

	• Boundary definition tools.
	• Risk analysis tools:
	  ○ Matrix of probability and impact.
	  ○ Analyze the pattern and impact of failure.
	  ○ Calculate the expected cash value.
	• Time estimation tools:
	  ○ Gantt Chart.
	  ○ Timeline Table.
	  ○ Network Programming (Activity Network).
	• Cost estimation tools:
	  ○ Calculating Payback Period and Return on Investment (ROI).
	  ○ Calculate the net present value (NPV).
	  ○ Calculation of standards and relative weights (Score).	

5.	**Analysis and Design Tools:** ▪ Introduction. • Dataflow Diagram. • Use Case Diagram. • Sequence Diagram. • Decision Tree. • Decision Table. • Software Architecture. • Entity Relationship Diagram. • Database Schema. • Data Dictionary. • Algorithms. • GUI Design.	16
Total		48

D. Students Assessment Activities

No	Assessment Activities *	Assessment timing (In week no)	Percentage of Total Assessment Score
1.	Midterm Exam	Week7	20%
2.	Quizzes	Continuous	10%
3.	Project	Continuous	20%
4.	Participation	Continuous	2%
5.	Practical Assessment	Continuous	8%
6.	Final Exam	Week13	40%

*Assessment Activities (i.e., Written test, oral test, oral presentation, group project, essay, etc.)

E. Learning Resources and Facilities

1. References and Learning Resources

Essential References	• Systems Analysis and Design Methods by Jeffrey L. Whitten, et al.
Supportive References	○ Systems Analysis and Design by Kendall K. and Kendall J. ○ Seven Basic Software Development Life Cycle Methodologies by Martin. ○ Project Management: Process, Technology, and Practice by Ganesh Vaidyanathan.
Electronic Materials	Online resources will be provided during class lectures.
Other Learning Materials	N/A

2. Required Facilities and equipment

Items	Resources
Facilities (Classrooms, laboratories, exhibition rooms, simulation rooms, etc.)	Classroom included: 1. A projector connected to a PC, preferably with Internet access. 2. A vertical sliding board. 3. An equipped computer lab with at least 25 seats.
Technology equipment (Projector, smart board, software, etc.)	Computing resources (Projector, data show, Smart Board, software-EdrawMax-, etc.)
Other equipment (Depending on the nature of the specialty)	N/A

F. Assessment of Course Quality

Assessment Areas/Issues	Assessor	Assessment Methods
Effectiveness of teaching	Student	1. Students feedback (collected through surveys) as per university policy/procedure. 2. Teacher's Course report.
Effectiveness of student's assessment	Faculty	1. Review of Course Reports.

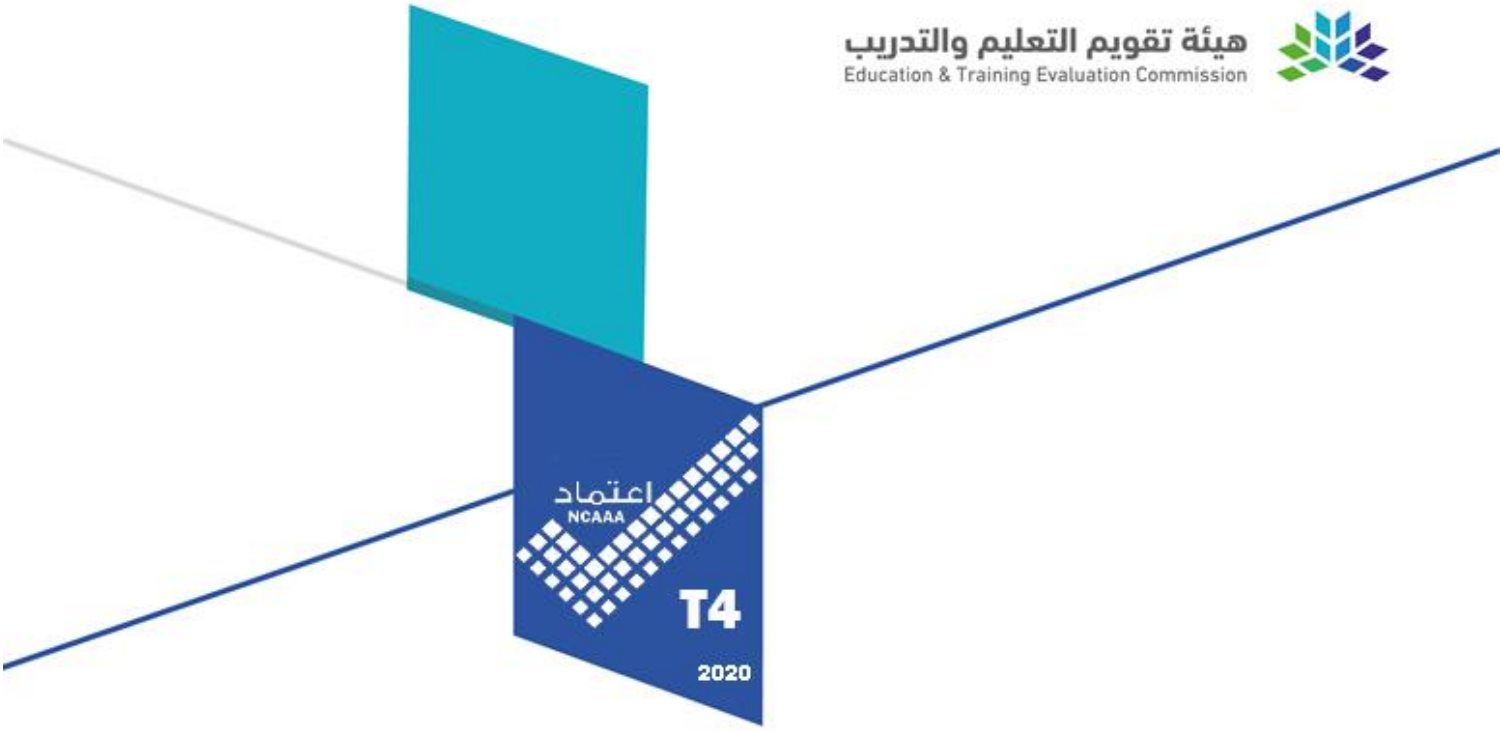
Assessment Areas/Issues	Assessor	Assessment Methods
		2. Review of Student feedback.
Quality of learning resources	Student and Faculty	Indirect using course evaluation and faculty survey.
The extent to which CLOs have been achieved	Program Leaders	Continuous review of the course contents, teaching strategies and utilizing the best practices.
Other	N/A	N/A

Assessor (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify)
Assessment Methods (Direct, Indirect)

G. Specification Approval Data

COUNCIL /COMMITTEE	
REFERENCE NO.	
DATE	

هيئة تقويم التعليم والتدريب
Education & Training Evaluation Commission



Course Specifications

Course Title:	Database Design
Course Code:	CS 230
Program:	Computer Science (Programming)
Department:	Applied Sciences
College:	Applied College
Institution:	Imam Muhammad Bin Saud Islamic University

Table of Contents

A. Course Identification

1. Credit hours: 3(2 theory , 2 lab)
2. Course type **a.** University ☐ College ☐ Department ☒ Others ☐ **b.** Required ☒ Elective ☐
3. Level/year at which this course is offered: 4th Semester
4. Pre-requisites for this course (if any)**:** CS 106 – CS 130
5. Co-requisites for this course (if any)**:** None

6. Mode of Instruction (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	**Traditional classroom**		
2	**Blended**	44	100 %
3	**E-learning**		
4	**Distance learning**		
5	**Other**		

7. Contact Hours (based on academic semester)

No	Activity	Contact Hours
1	**Lecture**	22
2	**Laboratory/Studio**	22
3	**Tutorial**	
4	**Others** (specify)	110
	Total	154

B. Course Objectives and Learning Outcomes

1. Course Description

This course deals with the basic concepts of databases and structured query language (SQL) and introduces the basics of building and designing different forms and screens. Over the course of the semester, the course includes an integrated case study in which all the previous tools are employed in addition to the experiences gained in the relevant courses to build an integrated project. This course is one of the courses that qualify for the following certificates:

- Database Administration Fundamentals SQL Server.
- Oracle Database 11g: SQL Fundamentals.
- Oracle Database SQL Certified Associate Certification Overview.

2. Course Main Objective

This course aims to introduce the student to the concept of databases, and how to use sentences for the structural query language, so that the student will be able at the end of the course to create a database using the phrases (DML) and (DDL) to create tables and add and modify data, where the student's knowledge of these topics is ensured through a software project delivered at the end of the semester. In addition to teaching the student the basics of designing different models in their basic form

3. Course Learning Outcomes

CLOs		Aligned PLOs
1	**Knowledge and Understanding**	
1.1	Familiarity with the basic concepts of Structured Query Language (SQL).	ع1، ع2، ع3، ع5
2	**Skills :**	
2.1	Be able to use DDL to deal with relational databases.	م1، م2، م4، م7
2.2	Proficiency in using DML when dealing with relational databases.	م1، م2، م4، م7
2.3	Mastery of dealing with tools of inventory, arrangement and linking in relational databases.	م1، م2، م4، م7
2.4	Control query sentences of various kinds in relational databases.	م1، م2، م4، م7
2.5	Use of information and communication technology in communication, exchange of ideas, scientific research, performance of tasks and costs.	م1، م2، م7
2.6	Practice critical thinking and solve problems facing the learner in the course in creative ways.	م1، م2، م7
3	**Values:**	
3.1	Cooperation, teamwork and professional ethics.	ق1
3.2	Take responsibility for continuous learning and continued personal development.	ق2
3.3	Manage time efficiently and effectively when applying the acquired knowledge and skills.	ق3

C. Course Content

No	List of Topics	Contact Hours
1	○ Introduction to Structured Query Language (SQL): • Review of the basics of databases (Database Management System- Database System) • Overview of Structured Query Language (SQL). • Development Stages of SQL language. • DBMS languages: ○ Data Definition Language (DDL) ○ Data Manipulation Language (DML) ○ Data Control Language (DCL)	2

2	○ Data Definition Language (DDL): • Create and delete a database. • Create, modify and delete tables. • Data handling. • Create, delete and modify field restrictions. • View databases, tables and fields.	8
3	○ Data Manipulation Language (DML): • Inserting records in tables. • Modification of records data. • Delete records from tables. • The general form of the query statement (Select). • Retrieve fields and columns with (Aliases). • Use calculations. • Priority of performing calculations. • Linking fields and columns. • Use the Distinct command to prevent duplicate records. • Dealing with the (null) value and its effects.	8
4	○ Enclosure and arrangement of data: • Select statement. • Where statement. • Requirements and guidelines for the condition clause. • Order statement (Order by). • Requirements and guidelines for the order statement. • Comparison factors used in the condition clause. • Other transactions used in condition clause. • Logical parameters. • Negation parameter (Not).	8
5	○ Using aggregate functions for more than one row: • Types of aggregate functions and how to use them. • Handle the count function. • Create sets of data using (Group by).	6
6	○ View data from more than one table: • Join definition. • Types of SQL JOINs: ○ Equal join ○ Unequal joins ○ Outer join. ○ Inner join for the same table. ○ Linking more than two tables.	4
7	○ Sub-query: • Types of queries. • Types of comparison parameters used with subquery. • Where to write sub-queries in Select statement. • Single-row subqueries. • Multi-row subqueries.	4

8	○ Multi-column subquery: • The general formula for the multi-column subquery statement. • The result of the main query when the subquery returns the (null) value. • Use subquery in (form) clause.	4
Total		44

D. Teaching and Assessment

1. Alignment of Course Learning Outcomes with Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
1.0	**Knowledge and Understanding**		
1.1	Familiarity with the basic concepts of Structured Query Language (SQL).	- Classroom lecture - Dialogue and discussion -Survey - Discovery learning - Self-learning - Developed lecture -Brainstorming - Web Polling - KWL Learning Schedule - Mind maps - Concept maps	- Traditional and electronic achievement tests - Classroom questions - Assignments and periodic evaluations - Presentations - Discussion and debate - Cognitive performance tests - Achievement file
2.0	**Skills**		
2.1	Be able to use DDL to deal with relational databases.	- Practical presentation - Developed lecture - Discovery learning - Peer learning - Self-learning - Dialogue and discussion - Web Polling -Brainstorming - Cooperative learning - Problem solving - Project-based learning - E-discussion forums	- Presentations - Grading scales - Performance tests - Production metrics - Observation - Software projects - Achievement file - Peer Evaluation - Self-evaluation
2.2	Proficiency in using DML when dealing with relational databases.		
2.3	Mastery of dealing with tools of inventory, arrangement and linking in relational databases.		
2.4	Control query sentences of various kinds in relational databases.		
2.5	Use of information and communication technology in communication, exchange of ideas, scientific research, performance of tasks and costs.		
2.6	Practice critical thinking and solve problems facing the learner in the course in creative ways.		
3.0	**Values**		
3.1	Cooperation, teamwork and professional ethics.	- Project-based learning - Cooperative learning	- Note cards - Discussion and dialogue - Classroom questions
3.2	Take responsibility for continuous learning and continued personal development.		

Code	Course Learning Outcomes	Teaching Strategies	Assessment Methods
3.3	Manage time efficiently and effectively when applying the acquired knowledge and skills.	- Dialogue and discussion - Practical lecture - Modeling and role models - Web Polling	- Grading scales - Measures of values - Self-evaluation - Peer Evaluation - Achievement file

2. Assessment Tasks for Students

#	Assessment task*	Week Due	Percentage of Total Assessment Score
1	Midterm	Week 6	20 %
2	(3-4 Quizzes) Quizzes	All Semester	10 %
3	Lab Evaluation	All Semester	15 %
4	Project	Week 10	10 %
5	Participation	All Semester	5 %
6	Final	Week 13	40 %
7	**Total**		100 %

***Assessment task** (i.e., written test, oral test, oral presentation, group project, essay, etc.)

E. Student Academic Counseling and Support

Arrangements for availability of faculty and teaching staff for individual student consultations and academic advice :

- Distribution of guidelines prepared by the Deanship of Admission and Registration Affairs.
- Allocating office hours for faculty members to follow up on students' academic inquiries and communicate via mail

Electronic and course page on the Blackboard system, activating discussions, support and providing feedback.

- Studying the academic problems of college students and everything related to the reasons for dismissal, academic stumbling, delay and low average.

F. Learning Resources and Facilities

1.Learning Resources

Required Textbooks	1. Introduction to SQL and PL/SQL – volume 1, ORACLE Corporation, (2111). 2. Oracle Developer: Build Forms I – volume 1, ORACLE Corporation, (2111).
Essential References Materials	-
Electronic Materials	The course website on the Internet through the Blackboard platform, through which assignments are submitted electronically, discussion questions are asked, and the course content is uploaded so that the student can return to lectures in full, publish electronic quizzes, and educational videos related to the course, and advertisements for the

	course are displayed and direct feedback is provided according to what the course professor sees.
Other Learning Materials	-

2. Facilities Required

Item	Resources
Accommodation (Classrooms, laboratories, demonstration rooms/labs, etc.)	• Classroom • Computer lab.
Technology Resources (AV, data show, Smart Board, software, etc.)	• Smart board • data projector • My SQL, SQL Server or PL/SQL.
Other Resources (Specify, e.g. if specific laboratory equipment is required, list requirements or attach a list)	-

G. Course Quality Evaluation

Evaluation Areas/Issues	Evaluators	Evaluation Methods
The effectiveness of teaching methods.	Peer References – Students	1. Questionnaires and referendums approved by the department. 2. Peer evaluation of faculty members. 3. Review the results of students' evaluation.
The effectiveness of student assessment methods.	Peer references – program leaders – faculty members – students	1. Questionnaires and referendums approved by the department. 2. Review course descriptions and course reports periodically. 3. Peer evaluation and periodic exchange of correction and scrutiny between faculty colleagues. 4. Review samples of students' work.
Learning Resources.	Program Leaders – Faculty Members – Students	1. Questionnaires and referendums approved by the department. 2. Write-off and monitoring lists.
The extent to which the learning outcomes of the course have been achieved.	Program Leaders – Faculty Members	1. Review the course report. 2. Analyze test forms, grades, student work and achievement records

Evaluation areas (e.g., Effectiveness of teaching and assessment, Extent of achievement of course learning outcomes, Quality of learning resources, etc.)
Evaluators (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify)
Assessment Methods (Direct, Indirect)

H. Specification Approval Data

Council / Committee	
Reference No.	
Date	