İZMİR UNIVERSITY OF ECONOMICS

Faculty of Engineering
Computer Engineering
SE 410 - Software Framework Applications

COURSE INTRODUCTION AND APPLICATION INFORMATION

Course Name	**Software Framework Applications**

Code	Semester	Theory (hour/week)	Application/Laboratory (hour/week)	Local Credits	ECTS
SE 410	Fall/Spring	3	0	3	5

Prerequisites	None
Course Language	English
Course Type	Elective
Course Level	First Cycle
Mode of Delivery	face to face
	* Problem Solving * Q&A * Lecture / Presentation
Course Coordinator	* Dr. Öğr. Üyesi Serhat UZUNBAYIR
Course Lecturer(s)	* Dr. Öğr. Üyesi Serhat UZUNBAYIR
Course Assistants	-

Course Objectives	The objective of this course is to present to students the concept of software framework and how to implement it for various purposes.
Course Learning Outcomes	The students who succeeded in this course; * Describe the concept of software framework. * Identify the 3-tier software architecture. * Discuss the protocols behind web services. * Utilize software framework to build applications.

	* Implement mechanisms and techniques for the deployment and configuration of .NET applications.
Course Description	This course introduces the students to the fundamental concepts of .NET framework application development using C# programming language. The course covers the major topics for .NET application programming over the .NET Framework. Via understanding these topics it will be very easy for the students to develop and deploy enterprise level applications via .NET framework.

Course Category	Core Courses	
	Major Area Courses	X
	Supportive Courses	
	Media and Managment Skills Courses	
	Transferable Skill Courses	

WEEKLY SUBJECTS AND RELATED PREPARATION STUDIES

Week16	Subjects	Related Materials
1	Review of the Semester	
2	Review of the Semester	
3	Review of the Semester	
4	Review of the Semester	
5	Review of the Semester	
6	Review of the Semester	
7	Review of the Semester	
8	Review of the Semester	
9	Review of the Semester	
10	Review of the Semester	
11	Review of the Semester	
12	Review of the Semester	
13	Review of the Semester	
14	Review of the Semester	

15	Review of the Semester	
16	Review of the Semester	

SOURCES

Course Notes / Textbooks	Programming C# 12: Build Cloud, Web, and Desktop Applications 1st Edition by Ian Griffiths
Suggested Readings/Materials	1) C# 12 and .NET 8 - Modern Cross-Platform Development Fundamentals - Eighth Edition: Start building websites and services with ASP.NET Core 8, Blazor, and EF Core 8 by Mark J Price 2) C# 12 in a Nutshell: The Definitive Reference 1st Edition by Joseph Albah

EVALUATION SYSTEM

Semester Activities	Number	Percentage of Grade
Participation	-	-
Laboratory / Application	-	-
Field Work	-	-
Quiz/Studio Critic	-	-
Portfoilo	-	-
Homework Assignment	-	-
Presentation/Jury	-	-
Project	1	30
Seminar/Workshop	-	-
Oral Exam	-	-
Midterm	1	30
Final	1	40
Total	3	100

WEIGHTING OF SEMESTER ACTIVITIES ON THE FINAL GRADE	**2**	**60**
WEIGHTING OF END-OF-SEMESTER ACTIVITIES ON THE FINAL GRADE	**1**	**40**
Total	3	100

ECTS / WORKLOAD TABLE

Semester Activities	Number	Duration (Hours)	Total Workload
Course Hours (Including Exam Week: 16 x Total Hours)	16	3	48
Laboratory / Application Hours	16	-	-
Study Hours Out of Class	14	3	42
Field Work	-	-	-
Quiz / Studio Critique	-	-	-
Portfolio	-	-	-
Homework / Assignment	-	-	-
Presentation / Jury	-	-	-
Project	1	20	20
Seminar / Workshop	-	-	-
Oral Exam	-	-	-
Midterm	1	20	20
Final	1	20	20
		Total Workload	**150**

THE RELATIONSHIP BETWEEN COURSE LEARNING OUTCOMES AND PROGRAM QUALIFICATIONS

#	Program Qualifications / Outcomes	* Level of Contribution				
		1	2	3	4	5
1	To have adequate knowledge in Mathematics, Science and Computer Engineering; to be able to use theoretical and applied information in these areas on complex engineering problems.					
2	To be able to identify, define, formulate, and solve complex Computer Engineering problems; to be able to select and apply proper analysis and modeling methods for this purpose.					
3	To be able to design a complex system, process, device or product under realistic constraints and conditions, in such a way as to meet the requirements; to be able to apply modern design methods for this purpose.					
4	To be able to devise, select, and use modern techniques and tools needed for analysis and solution of complex problems in Computer Engineering applications; to be able to use information technologies effectively.					
5	To be able to design and conduct experiments, gather data, analyze and interpret results for investigating complex engineering problems or Computer Engineering research topics.					
6	To be able to work efficiently in Computer Engineering disciplinary and multi-disciplinary teams; to be able to work individually.					
7	To be able to communicate effectively in Turkish, both orally and in writing; to be able to author and comprehend written reports, to be able to prepare design and implementation reports, to present effectively, to be able to give and receive clear and comprehensible instructions.					
8	To have knowledge about global and social impact of Computer Engineering practices on health, environment, and safety; to have knowledge about contemporary issues as they pertain to engineering; to be aware of the legal ramifications of Computer Engineering solutions.					
9	To be aware of ethical behavior, professional and ethical responsibility; to have knowledge about standards utilized in engineering applications.					
10	To have knowledge about industrial practices such as project management, risk management, and change management; to have awareness of entrepreneurship and innovation; to have knowledge about sustainable development.					
11	To be able to collect data in the area of Computer Engineering, and to be able to communicate with colleagues in a foreign language. ("European Language Portfolio Global Scale", Level B1)					
12	To be able to speak a second foreign language at a medium level of fluency efficiently.					

13	To recognize the need for lifelong learning; to be able to access information, to be able to stay current with developments in science and technology; to be able to relate the knowledge accumulated throughout the human history to Computer Engineering.					

*1 Lowest, 2 Low, 3 Average, 4 High, 5 Highest