

**Vocational School / Computer Programming**  
**2023 - 2024 Academic Year**  
**SOFTWARE DEVELOPER PLATFORMS and TOOLS**  
**Syllabus**

Course Description					
Name	Code	Semester	T+A Hour	Credit	ECTS
SOFTWARE DEVELOPER PLATFORMS and TOOLS	BPR2260530	Spring Semester	1+2	2	5
<b>Prerequisites Courses</b>					
<b>Recommended Elective Courses</b>					
<b>Language of Instruction</b>	Turkish				
<b>Course Level</b>	Short Cycle (Associate's Degree)				
<b>Course Type</b>	Elective				
<b>Course Coordinator</b>	Lect. Beyza KOYULMUŞ				
<b>Name of Lecturer(s)</b>	Lect. Özkan SARI				
<b>Assistant(s)</b>					
<b>Aim</b>	This course aims to introduce the concepts and principles of software engineering, software processes, software quality, software development methods, and the use of UML for system modeling.				
<b>Course Content</b>	This course contains; Software development concepts and integrated software development environments, Introduction to Configuration Management Systems and How to use GIT, Design (UML) Diagrams, Software development models and process steps, Software test techniques and unit tests, Software test tools, Requirement Engineering, Software project management and tools, Object oriented design and solution, Design models, Design modelling tools, Design process and principles, Software quality and its measures, Software quality tools.				
<b>Course Learning Outcomes</b>		<b>Teaching Methods</b>	<b>Assessment Methods</b>		
Defines various design principles		10, 12, 13, 14, 16, 17, 2, 4, 5, 8, 9	A, D, E, F, G		
Understands the importance of software development approach.		10, 12, 14, 2, 8	A, E, F		
Understands the basic concepts and principles of software development.		12, 14, 2, 6	A, E, F		
Understands software development processes.		12, 14, 2, 6, 8	A, E, F		
Defines software quality concepts and techniques.		10, 12, 14, 2, 6, 8, 9	A, E, F		
Knows various software development tools and techniques.		12, 14, 2, 6, 9	A, E, F		
Fundamental modeling using UML		12, 14, 2, 6, 8, 9	A, E, F		
<b>Teaching Methods</b>	10: Discussion Method, 12: Problem Solving Method, 13: Case Study Method, 14: Self Study Method, 16: Question - Answer Technique, 17: Experimental Technique, 2: Project Based Learning Model, 4: Inquiry-Based Learning, 5: Cooperative Learning, 6: Experiential Learning, 8: Flipped Classroom Learning, 9: Lecture Method				
<b>Assessment Methods</b>	A: Traditional Written Exam, D: Oral Exam, E: Homework, F: Project Task, G: Quiz				
<b>Lecture Schedule</b>					
<b>Sequence</b>	<b>Topics</b>	<b>Preliminary Preparation</b>			
1	Software development concepts and integrated software development environments				
2	Introduction to Configuration Management Systems and How to use GIT				
3	Design (UML) Diagrams				
4	Software development models and process steps.				
5	Software test techniques and unit tests				
6	Software test tools				
7	Requirement Engineering				
8	Software project management and tools				
9	Object oriented design and solution				
10	Design models				
11	Design modelling tools				
12	Design process and principles				
13	Software quality and its measures				
14	Software quality tools				
<b>Evaluation Methods</b>		<b>Weight(%)</b>			
Midterm Exam		40			
General Exam		60			

Resources	