

## COURSE DESCRIPTION FORM

### FAST-NUCES

INSTITUTION \_\_\_\_\_

PROGRAM (S) TO BE BSCS

EVALUATED \_\_\_\_\_

#### A. Course Description

(Fill out the following table for each course in your computer science curriculum. A filled out form should not be more than 2-3 pages.)

Course Code	CS-3004
Course Title	Software Design and Analysis
Credit Hours	3+0
Prerequisites by Course(s) and Topics	CS-309 Data Structures
Assessment Instruments with Weights (homework, quizzes, midterms, final, programming assignments, lab work, etc.)	Project + Presentation 10% Assignments 10% Midterms 30% Final 50%
Course Coordinator	Engr. Abdul Rahman
URL (if any)	<a href="http://slate.nu.edu.pk/portal/site/KHICS309FALL2019CS">http://slate.nu.edu.pk/portal/site/KHICS309FALL2019CS</a>
Current Catalog Description	Object Oriented approach, at present, is the method of choice for the industry to develop different software. It is a marked shift, in the way a software solution is conceived and implemented, from the structured/procedural design paradigm. Instead of viewing the problem domain as a sequence or set of procedures, the emphasis in OOA/D is on entities that interact with one another while making a design closer to the problem domain and the way human beings think and understand the real world.
Textbook (or Laboratory Manual for Laboratory Courses)	<ul style="list-style-type: none"><li>• UML 2 Toolkit by Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado</li><li>• UML and the Unified Process, Practical object-oriented analysis and design by Jim Arlow, Ila Neustadt</li></ul>
Reference Material	<ul style="list-style-type: none"><li>• Applying UML and Patterns 3rd Edition by Craig Larman</li><li>• The Unified Modeling Language Reference Manual, 2nd edition by James Rumbaugh, Ivar Jacobson and Grady Booch</li><li>• UML Distilled, 3rd Edition by Martin Flower</li><li>• Internet</li></ul>

Course Goals		A. Course Learning Outcomes (CLOs)			
CLO	Name	Domain	Taxonomy Level	Tools	
01	Understand the role of design and its major activities within the software development process, with focus on the Unified process.	Cognitive	1,2	A,M,F	
02	Comprehend the advantages of consistent and reliable software design.	Cognitive	1,2	A,M,F	
03	Design and Implement OOD models and refine them to reflect implementation details	Cognitive	3,4	A,M,F,P	
04	Apply and use UML to visualize and document the design of software systems.	Cognitive	3,4,5	A,M,F,P	
Tool: A = Assignment, M = Midterm, F=Final, P = Project					
B. Program Learning Outcomes					
For each attribute below, indicate whether this attribute is covered in this course or not. Leave the cell blank if the enablement is little or non-existent.					
PLO 1	Computing Knowledge	Apply knowledge of mathematics, natural sciences, computing fundamentals, and a computing specialization to the solution of complex computing problems.			✓
PLO 2	Problem Analysis	Identify, formulate, research literature, and analyze complex computing problems, reaching substantiated conclusions using first principles of mathematics, natural sciences, and computing sciences.			✓
PLO 3	Design/Develop Solutions	Design solutions for complex computing problems and design systems, components, and processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.			✓
PLO 4	Investigation & Experimentation	Conduct investigation of complex computing problems using research-based knowledge and research-based methods			
PLO 5	Modern Tool Usage	Create, select, and apply appropriate techniques, resources and modern computing tools, including prediction and modelling for complex computing problems.			✓
PLO 6	Society Responsibility	Apply reasoning informed by contextual knowledge to assess societal, health, safety, legal, and cultural issues relevant to context of complex computing problems.			
PLO 7	Environment and Sustainability	Understand and evaluate sustainability and impact of professional computing work in the solution of complex computing problems			
PLO 8	Ethics	Apply ethical principles and commit to professional ethics and responsibilities and norms of computing practice.			✓

	<b>PLO 9</b>	Individual and Team Work	Function effectively as an individual, and as a member or leader in diverse teams and in multi-disciplinary settings.										✓		
	<b>PLO 10</b>	Communication	Communicate effectively on complex computing activities with the computing community and with society at large.										✓		
	<b>PLO 11</b>	Project Mgmt and Finance	Demonstrate knowledge and understanding of management principles and economic decision making and apply these to one's own work as a member or a team.												
	<b>PLO 12</b>	Life Long Learning	Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological changes.												
<b>C. Relation between CLOs and PLOs</b> (CLO: Course Learning Outcome, PLOs: Program Learning Outcomes)															
		<b>PLOs</b>													
			<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>	
<b>CLOs</b>	1	✓													
	2		✓												
	3			✓											
	4					✓									
<b>Topics Covered in the Course, with Number of Lectures on Each Topic</b> (assume 15-week instruction and one-hour lectures)	<b>1. Topics to be covered:</b>														
	Weeks	List of Topics							No. of Weeks	Contact Hours	CLO				
	1	Overview of SAD & OOAD, SDLC, Software Environments, Flow charts DFDs, Intro to OOP (Java).							1	3	1,2				
	2	Domain Model, User Stories, Software Process Models & UP							1	3	1,2				
	3	Use case Diagrams							1	3	3,4				
	4	Class Diagrams, Class Relationships, Interfaces, Parameterized Class, Enumeration, Exceptions							1	3	3,4				
	5	Use-Case Realization, Entity, Control and Boundary classes							1	3	3,4				
	6	Mid Term 1													
	7	Activity Diagrams							1	3	3,4				
	8	UML for Real-Time Systems, Active Class / Object, Asynchronous / Synchronous Communication, Events / Triggers, Signals (Change / Signal / Call / Time trigger), Messages (Synchronous, Reply, Create, Asynchronous, Lost, Found), Synchronization and Concurrency, Fault Tolerance, Thread Implementation in Java,							1	3	1,2				

		Model, Views and Diagrams, 4+1 view model, Architectural views			
	9	Interaction Diagrams, Sequence and Collaboration Diagrams	1	3	3,4
	10	Timing Diagrams Homogenization of Classes	1	2	3,4 1,2
	11	Implementation, Component and Deployment Diagrams	1	1 2	1,2 3,4
	12	Mid Term 2			
	13	State Chart Diagrams and MVC	1	2 1	3,4 1,2
	14	Introductions to Design Patterns, Singleton Pattern, Facade	1	3	3,4
	15	Factory & Adapter Pattern	1	3	3,4
	16	Review	0.5	1.5	2,3,4
	17	Project Submission & Presentation	1.5	4.5	1,2,3,4
		<b>Total</b>	<b>15</b>	<b>45</b>	
	<b>Laboratory Projects/Experiments Done in the Course</b>	There will be class activities carried out after covering course topics in the form of case study evaluation and other class activities to help better learn the concepts. Tool will be Papyrus and Java will be the Language for OOP. There will be weekly tasks given to students.			
<b>Programming Assignments Done in the Course</b>	4				
<b>Class Time Spent on (in credit hours)</b>	<b>Theory</b>	<b>Problem Analysis</b>	<b>Solution Design</b>	<b>Social and Ethical Issues</b>	
	20	15	6	1	
<b>Oral and Written Communications</b>	Every student is required to submit at least __1__ written report of typically __2__ pages and to make __1__ oral presentations of typically __10__ minute's duration. Include only material that is graded for grammar, spelling, style, and so forth, as well as for technical content, completeness, and accuracy.				

**Instructor Name** Engr. Abdul Rahman Mahmood

**Instructor Signature** \_\_\_\_\_

**Date** 01-09-2021