DESCRIPTION/Syllabi of Curricula/Module

Short Name of the University/Country code **Date (Month / Year)**	**DSEA/P11** **Jan 2020**
TITLE OF THE MODULE	**Code**
Modern methods of designing software systems based on an object-oriented approach	

Teacher(s)	**Department**
Coordinating: Oleksandr Tarasov, Prof., Doctor of Sciences **Others:**	Department of Computer and Information Technology (CIT)

Study cycle **(BA/MA)**	**Level of the module** **(Semester number)**	**Type of the module** **(compulsary/elective)**
MA	2nd semester for Master	Elective

Form of delivery **(theory/lab/exercises)**	**Duration** **(weeks/months)**	**Language(s)**
Lectures, labs	15 weeks	Ukrainian / English

Prerequisites	
Prerequisites: study of disciplines: "Technology of software development", "Design and project management of information systems", "Modeling of systems", "Theory of computer-aided design".	**Co-requisites (if necessary):** Design skills, Programming skills

ECTS (Credits of the module)	Total student workload hours	Contact hours	Individual work hours
5	150	60	90

Aim of the module (course unit): competencies foreseen by the study programme

Students should be able to:

- to use the system analysis of objects and processes of computerization in systems of various function (in technical, organizational, medical systems), to design complex OOPS on the basis of OOPS methodology, use of design templates, UML, SysML, and technological process of OOPS design (UP) .
- to gain theoretical and practical skills of using design templates and to practically master the technology of OOPS design;
- to master the methods and techniques associated with the search for rational solutions in the design of OOPS for the automation of human activity in applied systems for various purposes, including in technical, organizational systems and in the field of medicine.

Learning outcomes of module (course unit)	Teaching/learning methods (theory, lab, exercises)	Assessment methods (written exam, oral exam, reports)
Knowledge: - methods of system analysis of objects and processes of computerization, design of complex OPS; - methodology of using UML, SysML, design templates in the design of OPS. This allows you to design and modify complex OPS based on the accumulated experience of programmers formalized in the form of design templates, UML, SysML and iterative technological process of OPS design. - theoretical and practical methods of creating conceptual, logical and physical models in the design of complex software systems; - methods and techniques associated with the search for rational solutions in the design of OPS; - types, purpose and characteristics of design templates and practical development of the technology of the iterative design process in the development and study of models of technical, organizational, technical and medical systems.	Work with the lecture notes as well as on the available fundamental subject literature	Written exam

Skills: - to formulate requirements for OOPS and technical systems based on interaction with experts, analysis of subject areas, search for analogs of models and software; - to develop models of subject areas of computerization using OOP methodologies, UML, SysML languages, modern mathematical apparatus, technologies for creating OOPS to ensure the effective operation of software and hardware; - to design and implement the information architecture of complex software systems for design tasks, modeling various objects and automating research in various subject areas, make informed design decisions in accordance with the requirements of customers, the capabilities of information technology, while using patterns for the distribution of duties, design patterns, experience gained in the field of OOPS development, recommendations for the implementation of the technological process of OOPS creation.	Lectures, project, consultation	Active attendance on lectures, individual/group project and presentation
Competences: ability to independently set and solve complex specialized tasks and scientific and practical problems in the field of computer science or in the process of learning about the creation of OOPS. To study subject literature, to exchange knowledge, to work in a group.	Lectures, project, consultation	Individual/group project and presentation

Themes	**Contact work hours**							**Time and tasks for individual work**	
	Lectures	Consultations	Seminars	Practiacl work	Laboratory work	Placements	**Total contact work**	**Individual work**	**Tasks**

1. Basic concepts of systems engineering. Model driven architecture® (MDA®). Meta Object Facility (MOF), MBSE	2						**2**	**12**	Study exam/ complete exercise
2. Introduction to SysML. UML, SysML and MOF languages. Definition of SysML. SysML charts, functionality. Minimal SysML specification. Example of SysML using:	4			2			**6**	**14**	Study exam/ complete exercise
3. Principles of construction of OOPS using patterns. Basic concepts of purpose and classification of design patterns.	2						**2**	**10**	Study exam/ complete exercise
4. Principles of duties distribution. Types, purpose and characteristics of responsibilities (GRASP). An example of developing a logical model of OOPS	6						**6**	**14**	Study exam/ complete exercise
5. GoF group design patterns. Areas of application of design patterns, characteristics of generating patterns: Examples of use of design patterns.	4			4			**8**	**14**	Study exam/ complete exercise
6. Characteristics and application of structural patterns. Examples of the use of structural design patterns. Mapping patterns.	4			4			**8**	**14**	Study exam/ complete exercise
7. Characteristics and application of behavior patterns. Examples of the use of behavior design patterns.	6			4			**10**	**14**	Study exam/ complete exercise
8. Iterative and incremental development of OOPS, tasks at different stages (phases) of the design process and iterations.	2			1			**3**	**8**	Study exam/ complete exercise
Total	**30**			**15**			**45**	**105**	

Assessment strategy	**Weight in %**	**Deadlines**	**Assessment criteria**

written exam theory	40%	during the semester / exam	Good response to the questions
Practical exam on a computer	60%	during the semester / exam	the work is done completely without mistakes or minor errors

Author	**Year of issue**	**Title**	**No of periodical or volume**	**Place of printing. Printing house or internet link**
Compulsory literature				
1. Robert C. Martin	2017	Clean Architecture : A Craftsman's Guide to Software Struc-ture and Design (1st Edition) 432 p.		Pearson Education ISBN10 0134494164 ISBN13 9780134494166
2. Martin Fowler	2018	Refactoring : Improving the Design of Existing Code 448 p		Addison-Wesley ISBN10 0134757599 ISBN13 9780134757599
3. Robert Martin	2009	Clean Code : A Handbook of Agile Software Craftsmanship (1st Edition) - 464 p		Pearson Education ISBN10 0132350882 ISBN13 9780132350884
4. Alan Dennis, Barbara Haley Wixom, & 1 more	2015	Systems Analysis and Design with UML (7th Edition) 464 p.		Wiley ISBN-10: 1119496489 ISBN-13: 978-1119496489
5. Grady Booch, Robert A. Maksimchuk, Michael W. Engle, & 3 more Буч Г. Объектно-ориентированный анализ и проектиро-вание с примерами приложений на C++, 2-е изд. /Пер. с англ. - М.: Бином: Невский диалект, 1999. – 560 с.	2007	Object-Oriented Analysis and Design with Applications (3rd Edition) -720 p.		Addison-Wesley ISBN-10: 020189551X ISBN-13: 978-0201895513
6. Craig Larman Ларман К. Применение UML и шаблонов проектирования. 2-е изд.: /Пер. с англ.- М.:	2004	Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and		Pearson ISBN-10: 0131489062

Издательский дом «Вильямс», 2002. – 624 с.		Iterative Development (3rd Edition) 736 p.		ISBN-13: 978-0131489066
7. Erich Gamma, Richard Helm , Ralph Johnson , John Vlissides Гамма Э. Приёмы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма, Р. Хелм, Р.Джонсон, Дж. Влиссидес. – СПб.: Питер, 2001. – 368 с.	1997	Design Patterns: Elements of Reusable Object-Oriented Software (1st Edition) -416 p.		Addison-Wesley ISBN-10: 0201633612 ISBN-13: 9780201633610
8. Eric Freeman, Elisabeth Robson	2020	Head First Design Patterns: Building Extensible and Maintainable Object-Oriented Software (2nd Edition) - 694 p.		O'Reilly Media ISBN10 149207800X ISBN13 9781492078005
Additional literature				
1. Grady Booch, Ivar Jacobson , et al. Буч Г., Рамбо Д., Джекобсон А. Язык UML. Руководство пользователя. - /Пер. с англ.- М.: ДМК, 2000. – 432 с.	1999	The Unified Modeling Language User Guide (Addison-Wesley Object Technology Series) 482 p.		Addison-Wesley ISBN-10: 0201571684 ISBN-13: 978-0201571684
2. Jim Arlow, Ila Neustadt	2005	UML 2 and the Unified Process: Practical Object-Oriented Analysis and Design (2nd Edition) 624 pages		Addison-Wesley ASIN : B0055O2HJI 10384 KB
3. Jim Conallen Коналлен Д. Разработка Web- приложений с использованием UML. /Пер с англ. – М.: Издательский дом «Вильямс», 2001. - 288 с.	2002	Building Web Applications with UML (2nd Edition) 496 p.		AddisonWesley Professional ISBN-10 9780201730388 ISBN-13: 978-0201730388
4. Ian Sommerville Соммервилл И. Инженерия программного обеспечения. - М.: Вильямс, 2002. -624 с.	2015	Software Engineering (10th Edition) - 816 p.		Pearson ISBN-10: 0133943038 ISBN-13: 978-0133943030

Holt J.	2008	SysML for systems engineering. - (Professional applications of computing series 7) Institution of Engineering and Technology -335 p.		ISBN 978-0-86341-825-9.
Friedenthal S., Moore A., Steiner. R.	2011	A Practical Guide to SysML: The Systems Modeling Language. 2nd edition, OMG Press. – 640 p.		ISBN-13: 978-0123852069. ISBN-10: 0123852064