WSB University						
Field of study:						
Course: Software Engineering lectures						
Educational profile: *practical*						
Education level: I-cycle studies						
Number of hours per semester	1		2		3	
	I	II	III	IV	V	VI
Full -time studies (L/C/lab/pr/e)*						
Part-time studies (L/C/lab/pr/e)						
CLASS LANGUAGE	English					
LECTURER	Rafał Deja					
FORM	Lecture					
COURSE OBJECTIVES	To familiarize students with issues related to the development and maintenance of complex IT systems. To present the fundamental principles and methods of client requirements analysis, design, coding, testing, and software maintenance.					

Reference to learning outcomes		**Description of learning outcomes**	**Verification of learning outcomes**
Field-related learning outcome	**PQF**		
Knowledge			
INF_W04	P6S_WG	The student knows and understands the engineering approach to software development.	Checkpoints in the online course / Exam
INF_W04, INF_W06	P6S_WG	The student knows and understands the iterative software development model.	Checkpoints in the online course / Exam
INF_W10	P6S_WG	The student knows and understands techniques and tools related to software analysis and design, as well as IT project management.	Checkpoints in the online course / Exam
Skills			
INF_U10	P6S_UW	The student can define project goals, decompose them and develop an implementation plan.	Checkpoints in the online course / Exam
INF_U01, INF_U10, INF_U15	P6S_UW	The student can collect and analyse requirements, prepare a requirements specification and design an IT system using UML diagrams.	Checkpoints in the online course / Exam
INF_U6	P6S_UW	The student can prepare the required documentation for the developed software product.	Checkpoints in the online course / Exam
Social competences			

INF_K02	P6S_KO	The student understands responsibility for the quality of developed software and the consequences of errors (technical and organizational); strives to work in accordance with good engineering practices.	Checkpoints in the online course / Exam
INF_K03	P6S_KR	The student demonstrates a lifelong learning attitude; is able to indicate areas requiring further knowledge acquisition in the context of the software development process and tools.	Checkpoints in the online course / Exam

Students' own workload (in didactic hours 1h=45 minutes)**

Full- time	**Part-time**
Participation in lectures =	Participation in lectures =
Participation in classes =	Participation in classes =
Preparation to classes =	Preparation to classes =
Preparation to lectures =	Preparation to lectures =
Preparation to an examination =	Preparation to an examination =
Project tasks =	Project tasks =
e-learning =	e-learning =
Credit/examination =	Credit/examination =
others (indicate which) =	others (indicate which) =
TOTAL:	**TOTAL:**
ECTS points:	**ECTS points:**
Including practical classes:	**Including practical classes:**

PREREQUISITES	Fundamentals of computer programming and object-oriented programming. Ability for abstract thinking. Knowledge of the basics of computer science and computer use. Fundamentals of mathematical logic.
COURSE CONTENT (Division into contact hours and e-learning)	**Contact hours:** • The software crisis and its consequences; why software engineering should be studied and what the scope of the course is (1 hour). • Software life-cycle models; advantages of the iterative approach and close cooperation with the client (1 hour). • Strategic phase: project preparation, cost estimation, planning with the use of MS Project, risk management (2 hours). • Requirements engineering, with particular emphasis on description through use cases and user stories (3 hours). • Version control systems (teamwork, change management) (1 hour) **E-learning:** • Object-oriented modelling and design using UML (4 hours). • Software quality assurance (2 hours). • Project management; traditional approach and agile/lightweight methodologies in software development management (2 hours)
LITERATURE (compulsory reading)	I. Somerville – Software Engineering, WNT 2003. https://www.computer.org/education/bodies-of-knowledge/software-engineering/v4

OPTIONAL LITERATURE (including at least two items in English, either books or articles)	Martin Fowler – UML Distilled: A Brief Guide to the Standard Object Modeling Language. Craig Larman – Applying UML and Patterns. Object-Oriented Analysis and Design and the Iterative Development. 3rd edition, Helion 2011.
SCHOLARLY PUBLICATIONS BY PERSONS WHO CONDUCT CLASSES, WHICH ARE RELATED TO THE MODULE SUBJECT	
TEACHING METHODS (Division into contact hours and e-learning)	**Contact hours:** Lecture using multimedia presentations. **E-learning:** Course online
TEACHING AIDS	
PROJECT (if implemented in the framework of a classes module)	Project goal: Topic of the project: Project form:
FORM AND CONDITIONS OF ASSESSMENT (Division into contact hours and e-learning)	Multiple-choice test exam.

** L-lecture, C- classes lab- laboratory, pro- project, e- e-learning*