DISCIPLINE SHEET

1. Program data

1.1 Higher education institution	Transylvania University of Brașov
1.2 Faculty	Mathematics and Computer Science
1.3 Department	Mathematics and Computer Science
1.4 Scopestudiesmaster's degree1)	Computer science
1.5 Study cycle2)	MASTERS
1.6 Study program/Qualification	INTERNET TECHNOLOGIES (IN ENGLISH)

2. Data about the discipline

2.1 Name of the discipline	Soft and Systems Quality Assurance							
2.2 Course activities holder				Lecturer Dr. Enache-David Nicoleta				
2.3 Seminar/laboratory/project activities holder				Lecturer Dr. Enache-David Nicoleta				
2.4 Year of study	yl	2.5 Semester	an d	2.6 Type of evaluation	It is	2.7 Discipline regime	Content3)	DAP
							Obligation4)	DI

3. Total estimated time (hours per semester of teaching activities)

3.1 Number of hours per week	**4**	of which: 3.2 course	**2**	3.3 seminar/laboratory/project	2
3.4 Total hours in the curriculum	56	of which: 3.5 course	28	3.6 seminar/laboratory/project	28
Distribution of time fund					hours
Study according to the textbook, course material, bibliography and notes					40
Additional documentation in the library, on specialized electronic platforms and in the field					40
Preparation of seminars/laboratories/projects, assignments, papers, portfolios and essays					26
tutorial					10
EXAMINATION					3
Other activities.................................					

3.7 Total student activity hours	119
3.8 Total hours per semester	175
3.9 Number of credits5)	7

4. Preconditions (where applicable)

4.1 curriculum	•
4.2 skills	•

5. Conditions (where applicable)

5.1 Course schedule	• Room equipped with video projector
5.2 of the seminar/laboratory/project	• Computer room

6. Specific skills acquired (according to the skills grid in the curriculum)

Professional skills	C1. Specification, design and development of software systems using: procedural languages, object-oriented languages, declarative languages, databases, methodologies and development platforms; R.Î. 1.1. The graduate can use procedural languages, object-oriented languages, declarative languages in dealing with theoretical and applied computer science issues; R.Î. 1.2. The graduate can frame a problem within a studied theoretical framework; R.O. 1.3. The graduate can apply modern programming methods and techniques to solve a wide range of problems; C3. Deepening the cutting-edge methodologies and technologies used in the software industry or with clear prospects of being used in the near future. R.Î.3.1. The graduate can present the historical evolution of the computer science concepts and theories in which he/she specialized; R.Î. 3.2. The graduate can convey well-organized computer knowledge to an audience; R.Î. 3.3. The graduate is able to make interconnections between different computer science fields; R.Î. 3.4. The graduate can produce synthesis materials on a theoretical or applied subject; R.O. 3.5. The graduate can frame a problem within a studied theoretical framework. C4. Establishes data processes, manages data collection systems, develops data processing applications, implements data quality processes, performs data extraction R.Î.4.1 The graduate uses ICT tools to apply mathematical, algorithmic or other data manipulation processes to create information. R.Î.4.2 The graduate develops and manages methods and strategies used to maximize data quality and statistical efficiency in data collection, to ensure that the data collected is optimized for further processing.
Transversal skills	CT.1. Communication and cooperation in professional contexts RI.1.1. The graduate uses a specific repertoire of communication with interlocutors belonging to different cultures, promoting intercultural communication. RI.1.2. The graduate uses communication and relationship techniques in the virtual environment. RI.1.3. The graduate is able to cooperate and integrate into professional work teams in the educational field and interdisciplinary teams. RI.1.4. The graduate adapts his/her language and communication repertoire to the particularities of the interlocutors. RI.1.5. The graduate can give presentations and public communications to promote knowledge and professional values. CT. 2. Career development and management RI.2.1. The graduate documents and identifies opportunities for continuing professional training. RI.2.2. The graduate formulates objectives regarding career development and identifies action strategies in this regard. RÎ.2.3. The graduate self-evaluates and reflects on his or her own career, identifying strategies for regulating and overcoming professional difficulties. RÎ.2.4. The graduate has strategies for regulating and controlling professional and personal stress. RÎ. 2.5. The graduate knows and applies professional and personal time management techniques.

7. Objectives of the discipline (based on the specific skills acquired)

7.1 General objective of the discipline	• Knowledge and application of concepts, principles and techniques for developing software systems • Training and development of skills in the use of software engineering support techniques
7.2 Specific objectives	• Knowledge of Web project development models • Developing skills in using modeling languages in the analysis and design of complex computer systems • Knowledge of the management stages of a software project

8. Contents

8.1 Course	Teaching methods	Number of hours	Observations
1. Introduction	The lecture, the explanation	2	
2. Testing programs	The lecture, the explanation	2	
3. Defects in software systems. Code inspection.	The lecture, the explanation	2	
4. Risk analysis. Test plans	The lecture, the explanation	2	

5. Testing levels: unit testing, integration testing, system testing, acceptance testing.	The lecture, the explanation	2	
6. Extreme testing.	The lecture, the explanation	2	
7. Regression testing	The lecture, the explanation	2	
8. Assertions. Debugging programs	The lecture, the explanation	2	
9. Quality measurement. Software quality metrics. Defect elimination	The lecture, the explanation	2	
10. Software Reliability Models	The lecture, the explanation	2	
11. Process metrics for testing	The lecture, the explanation	2	
12. Complexity metrics	The lecture, the explanation	2	
13. Complexity metrics	The lecture, the explanation	2	
14.Recap	The lecture, the explanation	2	

Bibliography
RD Craig, SP Jaskiel, Systematic Software Testing, SQE Publishing, 2007
SH Kahn, Metrics and Models in Software Quality Engineering, Second Edition, Addison-Wesley, 2003
Robert V. Binder, Testing Object-Oriented Systems: Models, Patterns, and Tools, Addison-Wesley, 2000
GJ Myers, The Art of Software Testing, Second Edition, Wiley, 2004

8.2 Seminar/laboratory/project	Teaching-learning methods	Number of hours	Observations
1. Testing programs	Exposition, exemplification	4	
2. Risk analysis. Test plans	Exposition, exemplification	4	
3. Quality measurement. Metrics for software quality. Defect elimination	Exposition, exemplification	8	
4. Software reliability models	Exposition, exemplification	8	
5. Complexity metrics	Exposition, exemplification	4	

Bibliography
RD Craig, SP Jaskiel, Systematic Software Testing, SQE Publishing, 2007
SH Kahn, Metrics and Models in Software Quality Engineering, Second Edition, Addison-Wesley, 2003
Robert V. Binder, Testing Object-Oriented Systems: Models, Patterns, and Tools, Addison-Wesley, 2000
GJ Myers, The Art of Software Testing, Second Edition, Wiley, 2004

9. Correlating the content of the discipline with the expectations of representatives of epistemic communities, professional associations and representative employers in the field related to the program

The contents are consistent with the issues of monographs published in recent years in the specialized literature.

10. Evaluation

Type of activity	10.1 Evaluation criteria	10.2 Evaluation methods	10.3 Weight of the final grade
10.4 Course	The ability to use the concepts presented in the analysis and testing stages is tested.	Written work – grid test	50%
10.5 Seminar/laboratory/project	Individual project presentation	Correct drafting of the project, in accordance with specific requirements	50%
10.6 Minimum performance standard			

- the ability to use the concepts presented in the analysis and design stages of a web application
- the quality of the project presentation

This Discipline Sheet was approved in the Department Council meeting on 09/26/2024 and approved in the Faculty Council meeting on 09/26/2024.

Associate Professor Dr. Ion Gabriel Stan Dean	Assoc. Prof. Dr. Nicușor Minculete Department manager
Lecturer Nicoleta Enache-David, PhD Course holder	Lecturer Nicoleta Enache-David, PhD Seminar/laboratory/project leader

Note:

[1] Field of study - choose one of the options: Bachelor's/Master's/Doctorate (is completed in accordance with the Nomenclature of fields and specializations/university study programs in force);

[2] Study cycle - choose one of the options: Bachelor's/Master's/Doctorate;

[3] Discipline regime (content) - choose one of the options: DF(fundamental discipline)/DD(discipline in the field)/DS(specialized discipline)/AD(complementary discipline) - for the bachelor's level;DAP (specialization discipline)/ISD(synthesis discipline)/DC underscored(advanced knowledge discipline) - for the master's level;

[4] Discipline regime (compulsory) - choose one of the options:DI (mandatory subject)/DO(optional subject)/DFac (optional subject);

[5] One credit is equivalent to 25 hours of study (teaching activities and individual study).