SYLLABUS

1. Data about the program of study

1.1	Institution	The Technical University of Cluj-Napoca
1.2	Faculty	Automation and Computer Science
1.3	Department	Computer Science
1.4	Field of study	Computer Science and Information Technology
1.5	Cycle of study	Bachelor of Science
1.6	Program of study/Qualification	Computer Science/ Engineer
1.7	Form of education	Full time
1.8	Subject code	50.

2. Data about the subject

2.1	Subject name					Information Systems					
2.2	Subject area					Computer Science and Information Technology					
2.3	Course responsible/lecturer					Assoc. prof. dr. eng. Ovidiu Pop – Ovidiu.Pop@cs.utcluj.ro					
2.4	Teachers in charge of applications					Assoc. prof. dr. eng. Ovidiu Pop – Ovidiu.Pop@cs.utcluj.ro					
2.5	Year of study	IV	2.6	Semester	8	2.7	Assessment	exam	2.8	Subject category	DS/OB

3. Estimated total time

Sem.	Subject name	Lecture	Applications			Lecture	Applications			Individual study	TOTAL	Credit
		[hours / week.]				[hours / semester]						
			S	L	P		S	L	P			
8	**Information Systems**	**2**	**-**	**2**	**-**	**28**	**-**	**28**	**-**	**47**	**103**	**4**

3.1	Number of hours per week	4	3.2	of which, course	2	3.3	applications	2
3.4	Total hours in the teaching plan	56	3.5	of which, course	28	3.6	applications	28

Individual study	Hours
Manual, lecture material and notes, bibliography	20
Supplementary study in the library, online and in the field	10
Preparation for seminars/laboratory works, homework, reports, portfolios, essays	5
Tutoring	
Exams and tests	5
Other activities	7

3.7	Total hours of individual study	47
3.8	Total hours per semester	103
3.9	Number of credit points	4

4. Pre-requisites (where appropriate)

4.1	Curriculum	Software engineering, database design
4.2	Competence	Object-oriented design

5. Requirements (where appropriate)

5.1	For the course	50% (attendance)
5.2	For the applications	80% (attendance)

6. Specific competences

Professional competences	**C4 -** Improving the performances of the hardware, software and communication systems (1 credit) **C4.1** - Identifying and describing the defining elements of the performances of the hardware, software and communication systems **C4.2** - Explaining the interaction of the factors that determine the performances of the hardware, software and communication systems **C4.3 -** Applying the fundamental methods and principles for increasing the performances of the hardware, software and communication systems **C4.4** - Choosing the criteria and evaluation methods of the performances of the hardware, software and communication systems **C4.5** - Developing professional solutions for hardware, software and communication systems based on performance optimization **C5 -** Designing, managing the lifetime cycle, integrating and ensuring the integrity of hardware, software and communication systems (1 credit) **C5.1** - Specifying the relevant criteria regarding the lifetime cycle, quality, security and the computing system's interaction with the environment and the human operator **C5.2** - Using interdisciplinary knowledge for adapting the computing system to the specific requirements of the application field **C5.3** - Using fundamental principles and methods for ensuring the security, the safety and ease of exploitation of the computing systems **C5.4** - Proper utilization of the quality, safety and security standards in the field of information processing **C5.5** - Creating a project including the problem's identification and analysis, its design and development, also proving an understanding of the basic quality requirements **C6 -** Designing intelligent systems (2 credits) **C6.1** - Describing the components of intelligent systems **C6.2** - Using domain-specific tools for explaining and understanding the functioning of intelligent systems **C6.3** - Applying the fundamental methods and principles for specifying solutions for typical problems using intelligent systems **C6.4** - Choosing the criteria and evaluation methods for the quality, performances and limitations of intelligent systems **C6.5** - Developing and implementing professional projects for intelligent systems
Cross competence	N/A

7. Discipline objectives (as results from the *key competences gained*)

7.1	General objective	Improve requirements management and design abilities of students in their senior year.
7.2	Specific objectives	Apply RUP metholologies for requirements management and design patterns

8. Contents

8.1. Lecture (syllabus)		Teaching methods	Notes
1	Requirements Maturity Management		
2	RUP – Overview and Best Practices		
3	RUP –Iterative Development		
4	The Requirements Discipline		
5	Capturing Requirements: Use Cases (I)		
6	Capturing Requirements: Use Cases (II) – Best Practices		
7	Analysis Model Artifacts: Vision, Glossary, Supplementary Specification (I)		
8	Analysis Model Artifacts: Vision, Glossary, Supplementary Specification (II)		
9	Domain Model		
10	GRASP Design Patterns (I)		
11	GRASP Design Patterns (II)		

12	Use Case Realizations with GRASP Design Patterns (I)		
13	Use Case Realizations with GRASP Design Patterns (II)		
14	Use Case Realizations with GRASP Design Patterns (III)		
Bibliography 1. Craig Larman – Applying UML and Patterns (2003) 2. Alistair Cockburn – Writing Effective Use Cases (2002)			
8.2. Applications (Laboratory)		Teaching methods	Notes
1	Requirements Artifacts: Vision, Glossary, Supplementary Specification	Students are encouraged to use their knowledge in implementation projects	
2	Generate a Vision document based on a RUP template		
3	Generate a Supplementary Specification document based on a RUP template		
4	Requirements Artifacts: Use Cases		
5	Generate a Use Case document based on a RUP template		
6	Generate an Analysis Model		
7	Lab Assessment		
Bibliography 1. Keneth Rubin – Essential Scrum (2012)			

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

The knowledge gained overlapping demands of all IT employers.

10. Evaluation

Activity type	10.1	Assessment criteria	10.2	Assessment methods	10.3	Weight in the final grade
Course				Written exam		80%
Applications				Problem solving		20%
10.4 Minimum standard of performance						
Proven understanding of requirements artifacts and ability to generate a design model.						

Course responsible
Assoc. prof. dr. eng. Ovidiu Pop

Head of department
Prof.dr.eng. Rodica Potolea

SYLLABUS

1. **Data about the program of study**

1.1	Institution	The Technical University of Cluj-Napoca
1.2	Faculty	Automation and Computer Science
1.3	Department	Computer Science
1.4	Field of study	Computer Science and Information Technology
1.5	Cycle of study	Bachelor of Science
1.6	Program of study/Qualification	Computer Science/ Engineer
1.7	Form of education	Full time
1.8	Subject code	51.1

2. **Data about the subject**

2.1	Subject name					Knowledge-Based Systems					
2.2	Subject area					Computer Science and Information Technology					
2.3	Course responsible/lecturer					Assoc.prof. dr. eng. Adrian Petru Groza – Adrian.Groza@cs.utcluj.ro					
2.4	Teachers in charge of applications					Lect. dr. eng. Anca Marginean Anca.Marginean@cs.utcluj.ro					
2.5	Year of study	IV	2.6	Semester	8	2.7	Assessment	exam	2.8	Subject category	DS/OP

3. **Estimated total time**

Sem.	Subject name	Lecture	Applications			Lecture	Applications			Individual study	TOTAL	Credit
		[hours / week.]				[hours / semester]						
			S	L	P		S	L	P			
8	**Knowledge-Based Systems**	**2**	**-**	**2**	**-**	**28**	**-**	**28**	**-**	**47**	**103**	**4**

3.1	Number of hours per week	4	3.2	of which, course	2	3.3	applications	2
3.4	Total hours in the teaching plan	56	3.5	of which, course	28	3.6	applications	28
Individual study								Hours
Manual, lecture material and notes, bibliography								28
Supplementary study in the library, online and in the field								14
Preparation for seminars/laboratory works, homework, reports, portfolios, essays								5
Tutoring								
Exams and tests								
Other activities								

3.7	Total hours of individual study	47
3.8	Total hours per semester	103
3.9	Number of credit points	4

4. **Pre-requisites (where appropriate)**

4.1	Curriculum	Introduction to Artificial Intelligence, Intelligent Systems
4.2	Competence	Important material that you should have learned: first order logic, algorithm design, big-O complexity analysis, heuristic search, logic programming, machine learning, formal verification methods. Useful skills that you should have: Linux, Latex, Java, LISP and Prolog programming languages.

5. **Requirements (where appropriate)**

5.1	For the course	Each student is required to enrol on moodle platform. By enrolling in this course, each student assumes the responsibility of an active participant in lecture and applications.
5.2	For the applications	

6. Specific competences

Professional competences	**C3** - Problems solving using specific Computer Science and Computer Engineering tools (1 credit) **C3.1** Identifying classes of problems and solving methods that are specific to computing systems **C3.2** Using interdisciplinary knowledge, solution patterns and tools, making experiments and interpreting their results **C3.3** Applying solution patterns using specific engineering tools and mehods **C3.4** Comparatively and experimentaly evaluation of the alternative solutions for performance optimization **C3.5** Developing and implementing informatic solutions for concrete problems **C5 -**Designing, managing the lifetime cycle, integrating and ensuring the integrity of hardware, software and communication systems (1 credit) **C5.1** Specifying the relevant criteria regarding the lifetime cycle, quality, security and computing system's interaction with the environment and human operator **C5.2** Using interdisciplinary knowledge for adapting the computing system to the specific requirements of the application field **C5.3** Using fundamental principles and methods for security, reliability and usability assurance of computing systems **C5.4** Adequate utilization of quality, safety and security standards in information processing **C5.5** Creating a project including the problem's identification and analysis, its design and development, also proving an understanding of the basic quality requirements **C6 -** Designing intelligent systems (2 credits) **C6.1** Describing the components of intelligent systems **C6.2** Using domain-specific tools for explaining and understanding the functioning of intelligent systems **C6.3** Applying the fundamental methods and principles for specifying solutions for typical problems using intelligent **C6.4** Choosing the criteria and evaluation methods for the quality, performances and limitations of intelligent systems **C6.5** Developing and implementing professional projects for intelligent systems
Cross competences	N/A

7. Discipline objectives (as results from the *key competences gained*)

7.1	General objective	Understanding conceptual instrumentation for knowledge representation and reasoning
7.2	Specific objectives	Applying various knowledge-based techniques aiming to increase the quality of software systems

8. Contents

8.1. Lecture (syllabus)		Teaching methods	Notes
1	Introduction: application case analysis, representative scenarios from different domains.	Slides, Warm-up examples, Quick individual work, Open discussions, Assignments, Round-up quizes	
2	Basic description logics: concepts, roles, instances, expressivity.		
3	Reasoning in description logics. Tableaux-based algorithms		
4	Description Logic Programs.		
5	Ontologies: formalisms, Semantic Web		
6	Ontology engineering: ontology design and evaluation		
7	Midterm assessment		
8	Rule-based systems: representation, reasoning methods.		
9	Non-monotonic reasoning		
10	Fuzzy systems: fuzzy sets, fuzzy inference, fuzzy expert systems		
11	Reasoning on knowledge: knowledge representation, epistemic logics		
12	Knowledge acquisition: conceptual knowledge, data mining, clustering.		
13	Model checking: computation tree logic		

14	Student presentation: ontology building competition		

Bibliography

- F. Baader, W. Nutt, Basic Description Logics, Handbook of Description Logics, Cambridge University Press, May 20, 2010
- Grosof, Benjamin N., et al. "Description logic programs: Combining logic programs with description logic." *Proceedings of the 12th international conference on World Wide Web*. ACM, 2003.
- Grigoris Antoniou and Frank van Harmelen, A Semantic Web Primer, second edition, MIT Press, 2008
- Horridge, Matthew, Bijan Parsia, and Ulrike Sattler. "Explaining inconsistencies in OWL ontologies." *Scalable Uncertainty Management*. Springer Berlin Heidelberg, 2009. 124-137.
- Andries P. Engelbrecht, Computational Intelligence An Introduction, second edition, Wiley, 2007
- Pang-Ning Tan, Michael Steinbach, Vipin Kumar, Introduction to data mining, Addison-Wesley, 2006
- Van Eijck and Verbrugge (eds.), Discourses on Social Software, Amsterdam University Press, 2009
- Michael Huth and Mark Ryan,Logic in Computer Science- Modelling and reasoning about systems 2000; Cambridge University Press, 2000
- Brachman, Ronald J., and Hector J. Levesque. "Knowledge representation and reasoning.." *Morgan Kaufmann Publishers,* 2004

8.2. Applications (Laboratory)		Teaching methods	Notes
1	Ontologies in KRSS syntax with RACER tool	Student engagement techniques, Examples, Deadlines	
2	Reusing ontologies		
3	Defining concepts		
4	Defining roles		
5	Populating ontologies		
6	Rules on top of ontologies		
7	Ontology design patterns		
8	Querying ontologies		
9	Integrating ontologies with other applications		
10	Debugging ontologies		
11	Ontology evaluation		
12	Documenting ontologies		
13	Ontology building competition		
14	Student presentations		

Bibliography

1. A. Groza - Ontology Engineering with RACER - an activity based approach, UTPress, 2014
2. Haarslev, Volker, and Ralf Möller. "RACER User s Guide and Reference Manual Version 1.7. 7." Concordia University and Univ. of Appl. Sciences in Wedel (2003).

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

Course organisation and its requiments follow the ACM guidelines and exemplary courses listed by ACM/IEEE Computer Science 2013 Exemplar-Fest
Employers in the field benefit from having a student more orietend towards increasing software quality.

10. Evaluation

Activity type	10.1	Assessment criteria	10.2	Assessment methods	10.3	Weight in the final grade
Course		Understanding conceptual instrumentation for knowledge representation and reasoning, Class participation, Assignments		Midterm assessment, Writing exam		70
Applications		Ontology evaluation metrics, Meeting deadlines, P\public		Lab project assessment		30

		presentation skills, Technical writing skills				
10.4 Minimum standard of performance:						
Understanding description logics, computational tree logic and rule-based systems. Meeting deadlines. Engineering a decent ontology.						

Course responsible
Assoc.prof.dr.eng. Adrian Groza

Head of department
Prof.dr.eng. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1	Institution	The Technical University of Cluj-Napoca
1.2	Faculty	Automation and Computer Science
1.3	Department	Computer Science
1.4	Field of study	Computer Science and Information Technology
1.5	Cycle of study	Bachelor of Science
1.6	Program of study/Qualification	Computer Science/ Engineer
1.7	Form of education	Full time
1.8	Subject code	51.2

2. Data about the subject

2.1	Subject name					Parallel Programming					
2.2	Subject area					Computer Science and Information Technology					
2.3	Course responsible/lecturer					Prof. dr. eng. Alin Suciu – alin.suciu@cs.utcluj.ro					
2.4	Teachers in charge of applications					Prof. dr. eng. Alin Suciu – alin.suciu@cs.utcluj.ro					
2.5	Year of study	IV	2.6	Semester	8	2.7	Assessment	exam	2.8	Subject category	DS/OP

3. Estimated total time

Sem.	Subject name	Lecture	Applications			Lecture	Applications			Individual study	TOTAL	Credit
		[hours / week.]				[hours / semester]						
			S	L	P		S	L	P			
8	**Parallel Programming**	**2**	**-**	**2**	**-**	**28**	**-**	**28**	**-**	**47**	**103**	**4**

3.1	Number of hours per week	4	3.2	of which, course	2	3.3	applications	2
3.4	Total hours in the teaching plan	56	3.5	of which, course	28	3.6	applications	28

Individual study	Hours
Manual, lecture material and notes, bibliography	18
Supplementary study in the library, online and in the field	12
Preparation for seminars/laboratory works, homework, reports, portfolios, essays	17
Tutoring	0
Exams and tests	0
Other activities	0

3.7	Total hours of individual study	47
3.8	Total hours per semester	103
3.9	Number of credit points	4

4. Pre-requisites (where appropriate)

4.1	Curriculum	Computer Programming (C), OO Programming (Java/C#), Logic Programming (Prolog), Operating Systems
4.2	Competence	All competences related to the above disciplines

5. Requirements (where appropriate)

5.1	For the course	Blackboard, Projector, Computer
5.2	For the applications	Multicore computers, Specific Software

6. Specific competences

Professional competences	**C3** - Problems solving using specific Computer Science and Computer Engineering tools (1 credit) **C3.1** Identifying classes of problems and solving methods that are specific to computing systems **C3.2** Using interdisciplinary knowledge, solution patterns and tools, making experiments and interpreting their results **C3.3** Applying solution patterns using specific engineering tools and mehods **C3.4** Comparatively and experimentaly evaluation of the alternative solutions for performance optimization **C3.5** Developing and implementing informatic solutions for concrete problems **C5 -**Designing, managing the lifetime cycle, integrating and ensuring the integrity of hardware, software and communication systems (1 credit) **C5.1** Specifying the relevant criteria regarding the lifetime cycle, quality, security and computing system's interaction with the environment and human operator **C5.2** Using interdisciplinary knowledge for adapting the computing system to the specific requirements of the application field **C5.3** Using fundamental principles and methods for security, reliability and usability assurance of computing systems **C5.4** Adequate utilization of quality, safety and security standards in information processing **C5.5** Creating a project including the problem's identification and analysis, its design and development, also proving an understanding of the basic quality requirements **C6 -** Designing intelligent systems (2 credits) **C6.1** Describing the components of intelligent systems **C6.2** Using domain-specific tools for explaining and understanding the functioning of intelligent systems **C6.3** Applying the fundamental methods and principles for specifying solutions for typical problems using intelligent **C6.4** Choosing the criteria and evaluation methods for the quality, performances and limitations of intelligent systems **C6.5** Developing and implementing professional projects for intelligent systems
Cross competences	N/A

7. Discipline objectives (as results from the *key competences gained*)

7.1	General objective	Developing the ability to identify parallelism in a given problem, and to take advantage of this parallelism using various methods and technologies for parallel programming
7.2	Specific objectives	▪ Understanding the performance parameters of parallel algorithms ▪ Ability to implement parallel algorithms using multithreading technologies (in C, Java, C#, Prolog, OpenMP) ▪ Ability to implement parallel algorithms based on the VSM model (Linda) ▪ Ability to implement parallel algorithms based on message passing (PVM, MPI) ▪ Basic knowledge of the cutting edge developments in the field (Quantum Computing, DNA Computing)

8. Contents

8.1. Lecture (syllabus)		Teaching methods	Notes
1	Introduction, Types of Parallelism, Classification, Applications	Lectures using blackboard and projector, interactive discussions	N/A
2	Parallel Algorithms, Performance Parameters, Amdahl's Law, Gustafson's Law		
3	Processes (C/UNIX), Communication, Synchronization		
4	Threads (Java, C#, Prolog), Communication, Synchronization		
5	OpenMP (1)		
6	OpenMP (2)		
7	OpenMP (3)		
8	Linda, Parallelism based on Virtual Shared Memory		
9	Message Passing Programming, PVM, MPI		

10	Programming the Graphics Processor (GPU)		
11	Sorting Networks		
12	Cryptography and Cryptanalysis concepts		
13	Grid Computing, Cluster Computing		
14	Quantum Computing and DNA Computing		

Bibliography

1. Peter Pacheco, An Introduction to Parallel Programming, Morgan Kaufmann, 2011.
2. Barbara Chapman, Gabriele Jost and Ruud van der Pas, Using OpenMP - Portable Shared Memory Parallel Programming, MIT Press, 2007 (online).
3. I. Foster, Designing and Building Parallel Programs, Addison Wesley, 1995 (online).
4. L. Sterling, E. Shapiro, The Art of Prolog, MIT Press, 1994.

8.2. Applications (Laboratory)		Teaching methods	Notes
1	Imperative Programming in C – review, Solving highly parallelizable problems	Practical laboratory works / programming exercises using specific software tools	N/A
2	Logic Programming in Prolog – review, Solving highly parallelizable problems		
3	Processes (C/UNIX)		
4	Threads (C)		
5	Threads (Java, C#)		
6	Threads (Prolog)		
7	Programming in OpenMP (1)		
8	Programming in OpenMP (2)		
9	Programming in OpenMP (3)		
10	Programming in Linda		
11	Programming in MPI		
12	Sorting Networks		
13	Cryptographic Algorithms		
14	Final Evaluation		

Bibliography

1. Peter Pacheco, An Introduction to Parallel Programming, Morgan Kaufmann, 2011.
2. Barbara Chapman, Gabriele Jost and Ruud van der Pas, Using OpenMP - Portable Shared Memory Parallel Programming, MIT Press, 2007 (online).
3. I. Foster, Designing and Building Parallel Programs, Addison Wesley, 1995 (online).
4. L. Sterling, E. Shapiro, The Art of Prolog, MIT Press, 1994.

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

The content of the course is aligned to the latest developments in the field and responds to both the development in the multicore / other parallel hardware technologies and the requirements coming from the industry.

10. Evaluation

Activity type	10.1	Assessment criteria	10.2	Assessment methods	10.3	Weight in the final grade
Course		Knowledge assimilated from the course material, interactivity during lectures		Written exam (E)		70 %
Applications		Ability to solve problem using parallel programming techniques and technologies		Laboratory assessment (L)		30 %
10.4 Minimum standard of performance						
E ≥ 50% and L ≥ 50%						

Course responsible
Prof.dr.eng. Alin Suciu

Head of department
Prof.dr.eng. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1	Institution	The Technical University of Cluj-Napoca
1.2	Faculty	Automation and Computer Science
1.3	Department	Computer Science
1.4	Field of study	Computer Science and Information Technology
1.5	Cycle of study	Bachelor of Science
1.6	Program of study/Qualification	Computer Science/ Engineer
1.7	Form of education	Full time
1.8	Subject code	52.1

2. Data about the subject

2.1	Subject name					DataBase Design					
2.2	Subject area					Computer Science and Information Technology					
2.3	Course responsible/lecturer					S.l. dr. ing. Călin Cenan – Calin.Cenan@cs.utcluj.ro					
2.4	Teachers in charge of applications					Conf. dr. ing. Delia Mitrea – Delia.Mitrea@cs.utcluj.ro					
2.5	Year of study	IV	2.6	Semester	8	2.7	Assessment	exam	2.8	Subject category	DS/OP

3. Estimated total time

Sem.	Subject name	Lecture	Applications			Lecture	Applications			Individual study	TOTAL	Credit
		[hours / week.]				[hours / semester]						
			S	L	P		S	L	P			
8	**DataBase Design**	**2**	**-**	**2**	**-**	**28**	**-**	**28**	**-**	**47**	**103**	**4**

3.1	Number of hours per week	4	3.2	of which, course	2	3.3	applications	2
3.4	Total hours in the teaching plan	56	3.5	of which, course	28	3.6	applications	28

Individual study	Hours
Manual, lecture material and notes, bibliography	12
Supplementary study in the library, online and in the field	20
Preparation for seminars/laboratory works, homework, reports, portfolios, essays	12
Tutoring	1
Exams and tests	2
Other activities	

3.7	Total hours of individual study	47
3.8	Total hours per semester	103
3.9	Number of credit points	4

4. Pre-requisites (where appropriate)

4.1	Curriculum	Database
4.2	Competence	

5. Requirements (where appropriate)

5.1	For the course	Board, video projector, computer; student present in mandatory 50% of days for admission to the final exam
5.2	For the applications	Computers, specific software; student present in mandatory 100% of days for admission to the final exam

6. Specific competences

Professional competences	**C3 -** Problems solving using specific Computer Science and Computer Engineering tools (2 credits) **C3.1 -** Identifying classes of problems and solving methods that are specific to computing systems **C3.2 -** Using interdisciplinary knowledge, solution patterns and tools, making experiments and interpreting their results **C3.3 -** Applying solution patterns using specific engineering tools and mehods **C3.4 -** Comparatively and experimentaly evaluation of the alternative solutions for performance optimization **C3.5 -** Developing and implementing information system solutions for concrete problems **C5** - Designing, managing the lifetime cycle, integrating and ensuring the integrity of hardware, software and communication systems (2 credits) **C5.1** - Specifying the relevant criteria regarding the lifetime cycle, quality, security and the computing system's interaction with the environment and the human operator **C5.2** - Using interdisciplinary knowledge for adapting the computing system to the specific requirements of the application field **C5.3** - Using fundamental principles and methods for ensuring the security, the safety and ease of exploitation of the computing systems **C5.4** - Proper utilization of the quality, safety and security standards in the field of information processing **C5.5** - Creating a project including the problem's identification and analysis, its design and development, also proving an understanding of the basic quality requirements
Cross competences	N/A

7. Discipline objectives (as results from the *key competences gained*)

7.1	General objective	Developing general skills in databases design and database programming
7.2	Specific objectives	Assimilate theoretical knowledge on relational databases design and SQL language extensions Presentation of database transactions Getting practical skills for designing and programming databases

8. Contents

8.1. Lecture (syllabus)		Teaching methods	Notes
1	Main steps to design a database; Data vs. Information	PDF & PPT Presentations; Demonstrations and model presentations on board; small exercises to increase interaction	
2	Historical roots of database ; Types of Databases		
3	Business Rules; Data Models: Hierarchical, Network, Relational, Entity-Relationship, Object Oriented		
4	Degrees of Data Abstraction; Conceptual Model; Internal Model; External Model; Physical Model		
5	Entity-Relationship concepts and terminology; Entity-Relationship diagrams; Tables; Keys, Attribute specifications; Data types; Data dictionary; Integrity constraints		
6	Relationships; Connectivity and Cardinality; Strength and Participation; Entity Supertypes and Subtypes		
7	Developing an ER Diagram; Optimization of Database – Normalization; Functional dependencies, 1NF, 2NF, 3NF, Boyce-Codd Normal Form (BCNF); 4NF, 5NF; Denormalization		
8	Constraints, Indexes; Data Definition Commands; Data Manipulation Commands		
9	Extended SQL – Transact-SQL; Writing Stored Procedures; Triggers		
10	Data / Information; Systems development life cycle: Planning, Analysis, Detailed Systems Design, Implementation		
11	Transaction Management and Concurrency Control; Transactions; Logs; Locks		
12	Data Warehouse - Need for Data Analysis; Decision Support Systems; Data Warehouse Architectures		
13	Facts, Dimensions, Attributes, Attribute Hierarchies; Data Mining		
14	Database administration; Security		

Bibliography

1. Alexandru Leluţiu - *Perenitatea Concepteleor Promovate de BAZELE de DATE*, Ed. Albastra, 2003

2. Raghu Ramakrishnan and Johannes Gehrke - *Database Management Systems*, McGraw-Hill Science, 2002
3. Peter Rob and Carlos Coronel - *Database Systems: Design, Implementation, and Management*, Crisp Learning, 2006
4. Rebecca M. Riordan - *Designing Relational Database Systems*, Microsoft Press, 1999
5. Matt Shepker - *Writing Stored Procedures for Microsoft SQL Server*, Sams, 2000
6. Mark Spenik and Orryn Sledge - *Microsoft SQL Server 2000 DBA Survival Guide*, Sams, 2001

8.2. Applications (Laboratory)		Teaching methods	Notes
1	Database and DataBase Management Systems - Microsoft SQL Server – Project domains	Exposure and applications	Computers, MS SQL Server, Oracle
2	Developing ER diagrams - Microsoft Visio		
3	Visio – SQL Server synchronization – First evaluation of project work: Domain analysis		
4	Design of Database Structures – Tables, Keys, Relationships		
5	Design of Database Structures – Indexes, Constraints, Views		
6	Update Data; Query Data - Second evaluation of project work: Database structures		
7	Simple Stored Procedures; Functions		
8	Stored Procedures - Cursors		
9	Triggers		
10	Transactions		
11	Data Warehouse		
12	Third evaluation of project work		
13	MS SQL Server administration		
14	Final laboratory evaluation - Final project evaluation		

Bibliography
1. Alexandru Leluţiu - *Perenitatea Concepteleor Promovate de BAZELE de DATE*, Ed. Albastra, 2003
2. Raghu Ramakrishnan and Johannes Gehrke - *Database Management Systems*, McGraw-Hill Science, 2002
3. Peter Rob and Carlos Coronel - *Database Systems: Design, Implementation, and Management*, Crisp Learning, 2006
4. Matt Shepker - *Writing Stored Procedures for Microsoft SQL Server*, Sams, 2000
5. Mark Spenik and Orryn Sledge - *Microsoft SQL Server 2000 DBA Survival Guide*, Sams, 2001

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

Database Design is an advanced topic in Computer Engineering and Information Technology field, combining fundamental aspects and practical software tools. Explaining to students the principles of database designing and database programming. Course content it is similar to database courses in other universities in the country and abroad.

10. Evaluation

Activity type	10.1	Assessment criteria	10.2	Assessment methods	10.3	Weight in the final grade
Course		Solving 4 problems and answers to questions of theory		2.5 hours written evaluation		60%
Applications		Implementarea unei aplicatii		Ongoing evaluation and a final presentation		40%

10.4 Minimum standard of performance

Solving practical laboratory work and projects, designing databases and a database programming; solving the problems and other subjects presented at the examination

Course responsible
Lect.dr.eng. Calin Cenan

Head of department
Prof.dr.eng. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1	Institution	The Technical University of Cluj-Napoca
1.2	Faculty	Automation and Computer Science
1.3	Department	Computer Science
1.4	Field of study	Computer Science and Information Technology
1.5	Cycle of study	Bachelor of Science
1.6	Program of study/Qualification	Computer Science/ Engineer
1.7	Form of education	Full time
1.8	Subject code	52.2

2. Data about the subject

2.1	Subject name					Computer Network Design					
2.2	Subject area					Computer Science and Information Technology					
2.3	Course responsible/lecturer					Assoc. Prof. dr. eng. Emil-Ioan Cebuc – Emil.Cebuc@cs.utcluj.ro					
2.4	Teachers in charge of applications					Lect. Dr. Eng. Bogdan Iancu – Bogdan.Iancu@cs.utcluj.ro					
2.5	Year of study	IV	2.6	Semester	8	2.7	Assessment	exam	2.8	Subject category	DS/OP

3. Estimated total time

Sem.	Subject name	Lecture	Applications			Lecture	Applications			Individual study	TOTAL	Credit
		[hours / week.]				[hours / semester]						
			S	L	P		S	L	P			
8	**Computer Network Design**	**2**	**-**	**2**	**-**	**28**	**-**	**28**	**-**	**47**	**103**	**4**

3.1	Number of hours per week	4	3.2	of which, course	2	3.3	applications	2
3.4	Total hours in the teaching plan	56	3.5	of which, course	28	3.6	applications	28

Individual study	Hours
Manual, lecture material and notes, bibliography	15
Supplementary study in the library, online and in the field	15
Preparation for seminars/laboratory works, homework, reports, portfolios, essays	12
Tutoring	2
Exams and tests	3
Other activities	0

3.7	Total hours of individual study	47
3.8	Total hours per semester	103
3.9	Number of credit points	4

4. Pre-requisites (where appropriate)

4.1	Curriculum	Local Area Networks, 7-th semester
4.2	Competence	LAN protocols, LAN structure, LAN services

5. Requirements (where appropriate)

5.1	For the course	Projector, Blackboard, lecture room
5.2	For the applications	PC with Linux/Windows OS, Switches, routers, hardware tools, cable tester

6. Specific competences

Professional competences	**C3 -** Problems solving using specific Computer Science and Computer Engineering tools (2 credits) **C3.1 -** Identifying classes of problems and solving methods that are specific to computing systems **C3.2 -** Using interdisciplinary knowledge, solution patterns and tools, making experiments and interpreting their results **C3.3 -** Applying solution patterns using specific engineering tools and mehods **C3.4 -** Comparatively and experimentaly evaluation of the alternative solutions for performance optimization **C3.5 -** Developing and implementing information system solutions for concrete problems **C5** - Designing, managing the lifetime cycle, integrating and ensuring the integrity of hardware, software and communication systems (2 credits) **C5.1** - Specifying the relevant criteria regarding the lifetime cycle, quality, security and the computing system's interaction with the environment and the human operator **C5.2** - Using interdisciplinary knowledge for adapting the computing system to the specific requirements of the application field **C5.3** - Using fundamental principles and methods for ensuring the security, the safety and ease of exploitation of the computing systems **C5.4** - Proper utilization of the quality, safety and security standards in the field of information processing **C5.5** - Creating a project including the problem's identification and analysis, its design and development, also proving an understanding of the basic quality requirements
Cross competences	N/A

7. Discipline objectives (as results from the *key competences gained*)

7.1	General objective	Knowledge and understanding of networking techniques, protocols and services
7.2	Specific objectives	Able to design simple network protocol at different OSI layer, able to configure networking devices at basic level

8. Contents

8.1. Lecture (syllabus)		Teaching methods	Notes
1	Introduction	Lecture, using PowerPoint presentation	
2	ISO-OSI an TCP/IP Reference models + Layered structure, analogies and differences		
3	Physical Layer + layer functions		
4	Data link Layer + layer function, HDLC protocol		
5	Network Layer + layer function and routing, IPv4 and IPv6		
6	Transport Layer + connection oriented and connection less protocols		
7	Upper Layers + session, presentation and application layers		
8	Multiplexing + FDM, TDM, statistical TDM		
9	Packet and circuit switching, virtual circuits + Analogies, differences and switches		
10	Flow control and congestion control + Stop and Wait, sliding window, token bucket		
11	Distributed network services like E-mail, DNS, etc.		
12	Network security + Threats and their avoidance		
13	Cryptographic systems+ symmetrical and asymmetrical systems		
14	Computer Network management + management application structure		

Bibliography

1. A. S. Tanenbaum, Computer Networks;
2. W. Stallings; Data and Computer Communications; Prentice Hall 2000

8.2. Applications (Laboratory)		Teaching methods	Notes
1	Sub netting and Super netting	Individual and team work Interactive	
2	Virtual LAN's VLAN		
3	Easy IP: DHCP,NAT		

4	DNS	tutoring Learn by example	
5	Static routing		
6	Dynamic routing		
7	Security		
8	Protocol Inspector II		
9	Network Inspector		
10	Application layer protocols		
11	Wireless I		
12	Wireless II		
13	Wireless IÎÎ		
14	Lab colloquium		

Bibliography

1. E. Cebuc et all, Computer Network Design Lab Guide, Editura UT Press 2005
2. Presentations can be found at: ftp.utcluj.ro/pub/users/cemil/prc

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

Course content is according to leading textbooks, lab content is inspired from CCNA industry certification level

10. Evaluation

Activity type	10.1	Assessment criteria	10.2	Assessment methods	10.3	Weight in the final grade
Course		Understands and explains network protocols, designs simple network Basic knowledge of network security and management		Written exam Problem solving Theory Admittance conditioned by successful lab colloquium		40% Theory 30% Problem
Applications		Is able to configure networking devices at basic level		Lab colloquium		30%
10.4 Minimum standard of performance						
Understands protocol stacks, flow and congestion control, network security and management issues. Configures switches and routers.						

Course responsible
Assoc.dr.eng. Emil Cebuc

Head of department
Prof.dr.eng. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1	Institution	The Technical University of Cluj-Napoca
1.2	Faculty	Automation and Computer Science
1.3	Department	Computer Science
1.4	Field of study	Computer Science and Information Technology
1.5	Cycle of study	Bachelor of Science
1.6	Program of study/Qualification	Computer Science/ Engineer
1.7	Form of education	Full time
1.8	Subject code	53

2. Data about the subject

2.1	Subject name					Project Management					
2.2	Subject area					Computer Science and Information Technology					
2.3	Course responsible/lecturer					Prof. dr. eng. Mihaela Dinsoreanu, mihaela.dinsoreanu@cs.utcluj.ro					
2.4	Teachers in charge of applications										
2.5	Year of study	IV	2.6	Semester	8	2.7	Assessment	exam	2.8	Subject category	DS/OB

3. Estimated total time

Sem.	Subject name	Lecture	Applications			Lecture	Applications			Individual study	TOTAL	Credit
		[hours / week.]				[hours / semester]						
			S	L	P		S	L	P			
8	**Project Management**	**2**	**-**	**-**	**-**	**28**	**-**	**-**	**-**	**49**	**77**	**3**

3.1	Number of hours per week	2	3.2	of which, course	2	3.3	applications	-
3.4	Total hours in the teaching plan	28	3.5	of which, course	28	3.6	applications	-

Individual study	Hours
Manual, lecture material and notes, bibliography	15
Supplementary study in the library, online and in the field	15
Preparation for seminars/laboratory works, homework, reports, portfolios, essays	3
Tutoring	13
Exams and tests	3
Other activities	

3.7	Total hours of individual study	49
3.8	Total hours per semester	77
3.9	Number of credit points	3

4. Pre-requisites (where appropriate)

4.1	Curriculum	Software Engineering
4.2	Competence	

5. Requirements (where appropriate)

5.1	For the course	Video projector (compulsory), internet connected computer (optional)
5.2	For the applications	-

6. Specific competences

Professional competences	**C5** Designing, managing the lifetime cycle, integrating and ensuring the integrity of hardware, software and communication systems **C5.1** Specifying the relevant criteria regarding the lifetime cycle, quality, security and the computing system's interaction with the environment and the human operator **C5.2** Using interdisciplinary knowledge for adapting the computing system to the specifc requirements of the application field **C5.3** Using fundamental principles and methods for ensuring the security, the safety and ease of exploitation of the computing systems **C5.4** Proper utilization of the quality, safety and security standards in the field of information processing **C5.5** Creating a project including the problem's identification and analysis, its design and development, also proving an understanding of the basic quality requirements
Cross competences	N/A

7. Discipline objectives (as results from the *key competences gained*)

7.1	General objective	Understand and apply appropriate project management techniques
7.2	Specific objectives	• Acknowledge the interfaces and interdependencies between the disciplines in OOSE • Present various project management techniques and their application in the two prominent methodologies • Project Management Metrics and Indicators • Understand the risks and the factors that lead to success or failure; Risk Management • Reflections of Project Management on the Software Quality

8. Contents

8.1. Lecture (syllabus)		Teaching methods	Notes
1	Introduction	Face to face lectures, Powerpoint slides	
2	PM overview		
3	Basics of Project Management for Agile Methodologies		
4	Basics of Project Management for Plan-driven Methodologies		
5	Planning and Tailoring the process		
6	Planning the Disciplines		
7	WBS development		
8	Time management		
9	Monitoring and Control		
10	Risk management		
11	Change management		
12	Resource management		
13	People management		
14	Project closure and final review		
Bibliography 5. Project Management Institute, A Guide to the Project Management Body of Knowledge, 5th Edition, 2013. 6. Juana Clark Craig, Project Management Lite: Just Enough to Get the Job Done…Nothing More, 2012 7. The Unified Software Development Process (Hardcover) Grady Booch, James Rumbaugh, Ivar Jacobson, Addison Wesley, 1998. 8. Software Project Management: A Unified Framework, Walker Royce, Addison Wesley 9. Planning Extreme Programming, Kent Beck, Addison Wesley, 2000			
8.2. Applications (Seminars, Laboratory, Projects)		Teaching methods	Notes
1			
Bibliography			

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

ACM Curriculum compliant course

10. Evaluation

Activity type	10.1	Assessment criteria	10.2	Assessment methods	10.3	Weight in the final grade
Course		Ability to apply appropriate PM techniques for given project situations		Written Exam		100%
Applications						
10.4 Minimum standard of performance						
Attendance of lectures >= 50% Grade of final exam >=5						

Course responsible
Prof.dr.eng. Mihaela Dinsoreanu

Head of department
Prof.dr.eng. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1	Institution	The Technical University of Cluj-Napoca
1.2	Faculty	Automation and Computer Science
1.3	Department	Computer Science
1.4	Field of study	Computer Science and Information Technology
1.5	Cycle of study	Bachelor of Science
1.6	Program of study/Qualification	Computer Science/ Engineer
1.7	Form of education	Full time
1.8	Subject code	54.

2. Data about the subject

2.1	Subject name					Project Elaboration Methodology					
2.2	Subject area					Computer Science and Information Technology					
2.3	Course responsible/lecturer					Conf.dr.ing.Tudor Muresan - Tudor.Muresan@cs.utcluj.ro					
2.4	Teachers in charge of applications					-					
2.5	Year of study	IV	2.6	Semester	8	2.7	Assessment	Colloquium	2.8	Subject category	DS/OB

3. Estimated total time

Sem.	Subject name	Lecture	Applications			Lecture	Applications			Individual study	TOTAL	Credit
		[hours / week.]				[hours / semester]						
			S	L	P		S	L	P			
8	**Project Elaboration Methodology**	**2**	**-**	**-**	**-**	**28**	**-**	**-**	**-**	**24**	**52**	**2**

3.1	Number of hours per week	2	3.2	of which, course	2	3.3	applications	-
3.4	Total hours in the teaching plan	28	3.5	of which, course	28	3.6	applications	-

Individual study	Hours
Manual, lecture material and notes, bibliography	18
Supplementary study in the library, online and in the field	4
Preparation for seminars/laboratory works, homework, reports, portfolios, essays	
Tutoring	
Exams and tests	4
Other activities	

3.7	Total hours of individual study	24
3.8	Total hours per semester	52
3.9	Number of credit points	2

4. Pre-requisites (where appropriate)

4.1	Curriculum	
4.2	Competence	

5. Requirements (where appropriate)

5.1	For the course	
5.2	For the applications	

6. Specific competences

Professional competences	**C5 -** Designing, managing the lifetime cycle, integrating and ensuring the integrity of hardware, software and communication systems **C5.1** - Specifying the relevant criteria regarding the lifetime cycle, quality, security and the computing system's interaction with the environment and the human operator **C5.2** - Using interdisciplinary knowledge for adapting the computing system to the specific requirements of the application field **C5.3 -** Using fundamental principles and methods for ensuring the security, the safety and ease of exploitation of the computing systems

	C5.4 - Proper utilization of the quality, safety and security standards in the field of information processing **C5.5** - Creating a project including the problem's identification and analysis, its design and development, also proving an understanding of the basic quality requirements
Cross competence	N/A

7. Discipline objectives (as results from the *key competences gained*)

7.1	General objective	1. Ability to write a project proposal 2. Ability to search literature and critical evaluation 3. Ability to use related work and technical reports 4. Ability to write literature reviews 4. Ability to write project documentation 5. Ability for oral presentation
7.2	Specific objectives	

8. Contents

8.1. Lecture (syllabus)		Teaching methods	Notes
1	Introduction - Computing project types	Using modern teaching methods and internet acces	
2	Choosing the project		
3	Preparing a project proposal		
4	Research and research process		
5	Research methods		
6	Literature search and review		
7	The report		
8	Structuring the report		
9	Writing the report		
10	Citing and reference management		
11	Reference styles		
12	Presenting and discussions on outstanding projects		
13	Oral presentation		
14	The talk and the defense		

Bibliography

1. Dawson, C.W. - Projects in Computing and Information Systems, Addison Wesley 2005
2. B. Olsson, M. Berndtsson, B. Lundell - Running Research-Oriented Final Year Projects for CS and IS Students, ACM SIGSE 2003
3. V. Bouki - Undergraduate Computer Science Projects in UK: What is the point?, Proc. of Informatics Education Europe II Conference, IEEII 2007

8.2. Applications (Seminars, Laboratory, Projects)		Teaching methods	Notes
1			

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

10. Evaluation

Activity type	10.1	Assessment criteria	10.2	Assessment methods	10.3	Weight in the final grade
Course				Colloquium		100%
Applications						
10.4 Minimum standard of performance						
Condition for credit achievement: N>=5						

Course responsible
Assoc.prof.dr.eng. Tudor Muresan

Head of department
Prof.dr.eng. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1	Institution	The Technical University of Cluj-Napoca
1.2	Faculty	Automation and Computer Science
1.3	Department	Computer Science
1.4	Field of study	Computer Science and Information Technology
1.5	Cycle of study	Bachelor of Science
1.6	Program of study/Qualification	Computer Science/ Engineer
1.7	Form of education	Full time
1.8	Subject code	55.

2. Data about the subject

2.1	Subject name					Communication protocols and networks project					
2.2	Subject area					Computer Science and Information Technology					
2.3	Course responsible/lecturer					Assoc. Prof. dr. eng. Emil Cebuc					
2.4	Teachers in charge of applications					Assoc.prof. dr. eng. Adrian Peculea, Lect. dr. eng. Bogdan Iancu					
2.5	Year of study	IV	2.6	Semester	8	2.7	Assessment	Colloquium	2.8	Subject category	DID/OB

3. Estimated total time

Sem.	Subject name	Lecture	Applications			Lecture	Applications			Individual study	TOTAL	Credit
		[hours / week.]				[hours / semester]						
			S	L	P		S	L	P			
8	**Communication protocols and networks project**	**-**	**-**	**-**	**2**	**-**	**-**	**-**	**28**	**24**	**52**	**2**

3.1	Number of hours per week	2	3.2	of which, course	-	3.3	applications	2
3.4	Total hours in the teaching plan	28	3.5	of which, course	-	3.6	applications	28

Individual study	Hours
Manual, lecture material and notes, bibliography	
Supplementary study in the library, online and in the field	24
Preparation for seminars/laboratory works, homework, reports, portfolios, essays	
Tutoring	
Exams and tests	
Other activities	

3.7	Total hours of individual study	24
3.8	Total hours per semester	52
3.9	Number of credit points	2

4. Pre-requisites (where appropriate)

4.1	Curriculum	Local Area Networks, 7-th semester
4.2	Competence	LAN protocols, LAN structure, LAN services

5. Requirements (where appropriate)

5.1	For the course	N/A
5.2	For the applications	Classroom, PC with internet access

6. Specific competences

Professional competences	**C5** Designing, managing the lifetime cycle, integrating and ensuring the integrity of hardware, software and communication systems (1 credit) **C5.1** Specifying the relevant criteria regarding the lifetime cycle, quality, security and the computing system's interaction with the environment and the human operator **C5.2 -** Using interdisciplinary knowledge for adapting an information system to application domain requirements **C5.3** Using fundamental principles and methods for ensuring the security, the safety and ease of exploitation of the computing systems **C5.4 -** Adequate utilization of quality, safety and security standards in information processing **C5.5** Creating a project including the problem's identification and analysis, its design and development, also proving an understanding of the basic quality requirements
Cross competences	**CT2** Identifying, describing and conducting processes in the projects management field, assuming different roles inside the team and clearly and concisely describing, verbally or in writing, in Romanian and in an international language, the results from the activity field. (1 credit)

7. Discipline objectives (as results from the *key competences gained*)

7.1	General objective	Teamwork, working with partial and contradicting specifications
7.2	Specific objectives	A team of 3-4 students is able to design a medium size LAN

8. Contents

8.1. Lecture (syllabus)		Teaching methods	Notes
1			
Bibliography			
8.2. Applications (Seminars, Laboratory, Projects)		Teaching methods	Notes
1	Introduction, team setup, project requirements and specifications	Brief presentation of possible solutions Refinement of project specifications	4 hours
2	Project design stage 1		4 hours
3	Project design stage 2		4 hours
4	Project design stage 3		4 hours
5	Project documentation 1		4 hours
6	Project documentation 2		4 hours
7	Project presentation and colloquium		4 hours
Bibliography 1. Packet Tracer user manual 2. OpNet user Manual 3. Equipment data sheet available on Internet, specific to each equipment selected by students			

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

Project content is kept state of the art by using latest devices available on the market .

10. Evaluation

Activity type	10.1	Assessment criteria	10.2	Assessment methods	10.3	Weight in the final grade
Course						
Applications		Submitted project fulfils requirements		Each project is evaluated individually		90% 10% activity during the face2face hours
10.4 Minimum standard of performance						
Students are able to select proper networking devices to fulfil design specifications. Students are able to configure equipment in a Packet Tracer simulation to fulfil specific functions.						

Course responsible
Assoc.prof.dr.eng. Emil Cebuc

Head of department
Prof.dr.eng. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1	Institution	The Technical University of Cluj-Napoca
1.2	Faculty	Automation and Computer Science
1.3	Department	Computer Science
1.4	Field of study	Computer Science and Information Technology
1.5	Cycle of study	Bachelor of Science
1.6	Program of study/Qualification	Computer Science/ Engineer
1.7	Form of education	Full time
1.8	Subject code	56.

2. Data about the subject

2.1	Subject name					Research and development activity					
2.2	Subject area					Computer Science and Information Technology					
2.3	Course responsible/lecturer					Diploma project supervisor					
2.4	Teachers in charge of applications					As decided by the supervisor					
2.5	Year of study	IV	2.6	Semester	8	2.7	Assessment	Verification	2.8	Subject category	DS/OB

3. Estimated total time

Sem.	Subject name	Lecture	Applications			Lecture	Applications			Individual study	TOTAL	Credit
		[hours / week.]				[hours / semester]						
			S	L	P		S	L	P			
8	**Research and development activity**	**-**	**-**	**-**	**8**	**-**	**-**	**-**	**112**	**122**	**234**	**9**

3.1	Number of hours per week	8	3.2	of which, course	-	3.3	applications	8
3.4	Total hours in the teaching plan	112	3.5	of which, course	-	3.6	applications	112

Individual study	Hours
Manual, lecture material and notes, bibliography	
Supplementary study in the library, online and in the field	120
Preparation for seminars/laboratory works, homework, reports, portfolios, essays	
Tutoring	
Exams and tests	2
Other activities	

3.7	Total hours of individual study	122
3.8	Total hours per semester	234
3.9	Number of credit points	9

4. Pre-requisites (where appropriate)

4.1	Curriculum	
4.2	Competence	

5. Requirements (where appropriate)

5.1	For the course	
5.2	For the applications	

6. Specific competences

Professional competences	**C4 -** Improving the performances of the hardware, software and communication systems (2 credits) **C4.1** - Identifying and describing the defining elements of the performances of the hardware, software and communication systems **C4.2** - Explaining the interaction of the factors that determine the performances of the hardware, software and communication systems **C4.3 -** Applying the fundamental methods and principles for increasing the performances of the hardware, software and communication systems **C4.4** - Choosing the criteria and evaluation methods of the performances of the hardware, software and communication systems **C4.5** - Developing professional solutions for hardware, software and communication systems based on performance optimization **C5 -** Designing, managing the lifetime cycle, integrating and ensuring the integrity of hardware, software and communication systems (2 credits) **C5.1** - Specifying the relevant criteria regarding the lifetime cycle, quality, security and the computing system's interaction with the environment and the human operator **C5.2** - Using interdisciplinary knowledge for adapting the computing system to the specific requirements of the application field **C5.3** - Using fundamental principles and methods for ensuring the security, the safety and ease of exploitation of the computing systems **C5.4** - Proper utilization of the quality, safety and security standards in the field of information processing **C5.5** - Creating a project including the problem's identification and analysis, its design and development, also proving an understanding of the basic quality requirements **C6 -** Designing intelligent systems (2 credit) **C6.1** - Describing the components of intelligent systems **C6.2** - Using domain-specific tools for explaining and understanding the functioning of intelligent systems **C6.3** - Applying the fundamental methods and principles for specifying solutions for typical problems using intelligent systems **C6.4** - Choosing the criteria and evaluation methods for the quality, performances and limitations of intelligent systems **C6.5** - Developing and implementing professional projects for intelligent systems
Cross competences	**CT1** Honorable, responsible, ethical behavior, in the spirit of the law, in order to ensure the professional reputation (1 credit) **CT2** Identifying, describing and conducting processes in the projects management field, assuming different roles inside the team and clearly and concisely describing, verbally or in writing, in Romanian and in an international language, the results from the activity field. (1 credit) **CT3** Demonstrating the spirit of initiative and action for updating professional, economical and organizational culture knowledge (1 credit)

7. Discipline objectives (as results from the *key competences gained*)

7.1	General objective	
7.2	Specific objectives	

8. Contents

8.1. Lecture (syllabus)		Teaching methods	Notes
1			
8.2. Applications (Projects)		Teaching methods	Notes
1	• Establish the topic of the diploma project • Establish the main chapters of the diploma thesis • Documentation on the topic of the diploma thesis • Write a synthesis of the bibliographic study		
Bibliography To be established by the supervisor of the diploma thesis.			

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

10. Evaluation

Activity type	10.1	Assessment criteria	10.2	Assessment methods	10.3	Weight in the final grade
Course						
Applications				The examination consists of the verification of the preliminary contents of the diploma work and the verification of the synthesis of the bibliographic study.		
10.4 Minimum standard of performance						

Course responsible
Diploma project supervisor

Head of department
Prof.dr.eng. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1	Institution	The Technical University of Cluj-Napoca
1.2	Faculty	Automation and Computer Science
1.3	Department	Computer Science
1.4	Field of study	Computer Science and Information Technology
1.5	Cycle of study	Bachelor of Science
1.6	Program of study/Qualification	Computer Science/ Engineer
1.7	Form of education	Full time
1.8	Subject code	57.

2. Data about the subject

2.1	Subject name					Practical placement for diploma thesis					
2.2	Subject area					Computer Science and Information Technology					
2.3	Course responsible/lecturer					Diploma project supervisor					
2.4	Teachers in charge of applications					As decided by the supervisor					
2.5	Year of study	IV	2.6	Semester	8	2.7	Assessment	Verification	2.8	Subject category	DS/OB

3. Estimated total time

Sem.	Subject name	Lecture	Applications			Lecture	Applications			Individual study	TOTAL	Credit
		[hours / week.]				[hours / semester]						
			S	L	P		S	L	P			
8	**Practical placement for diploma thesis**									**60**	**60**	**2**

3.1	Number of hours per week	-	3.2	of which, course	-	3.3	applications	-
3.4	Total hours in the teaching plan	-	3.5	of which, course	-	3.6	applications	-

Individual study	Hours
Manual, lecture material and notes, bibliography	20
Supplementary study in the library, online and in the field	8
Preparation for seminars/laboratory works, homework, reports, portfolios, essays	30
Tutoring	
Exams and tests	2
Other activities	

3.7	Total hours of individual study	60
3.8	Total hours per semester	60
3.9	Number of credit points	2

4. Pre-requisites (where appropriate)

4.1	Curriculum	
4.2	Competence	

5. Requirements (where appropriate)

5.1	For the course	
5.2	For the applications	

6. Specific competences

	C4 - Improving the performances of the hardware, software and communication systems (2 credits) **C4.1** - Identifying and describing the defining elements of the performances of the hardware, software and communication systems **C4.2** - Explaining the interaction of the factors that determine the performances of the hardware, software and communication systems **C4.3 -** Applying the fundamental methods and principles for increasing the performances of the hardware, software and communication systems

Professional competences	**C4.4** - Choosing the criteria and evaluation methods of the performances of the hardware, software and communication systems **C4.5** - Developing professional solutions for hardware, software and communication systems based on performance optimization **C5 -** Designing, managing the lifetime cycle, integrating and ensuring the integrity of hardware, software and communication systems (2 credits) **C5.1** - Specifying the relevant criteria regarding the lifetime cycle, quality, security and the computing system's interaction with the environment and the human operator **C5.2** - Using interdisciplinary knowledge for adapting the computing system to the specific requirements of the application field **C5.3** - Using fundamental principles and methods for ensuring the security, the safety and ease of exploitation of the computing systems **C5.4** - Proper utilization of the quality, safety and security standards in the field of information processing **C5.5** - Creating a project including the problem's identification and analysis, its design and development, also proving an understanding of the basic quality requirements
Cross competences	N/A

7. Discipline objectives (as results from the *key competences gained*)

7.1	General objective	Elaboration of the diploma thesis.
7.2	Specific objectives	To achieve these general objectives, students will integrate the research results in a paper to comply with the requirements of the department.

8. Contents

8.1. Lecture (syllabus)		Teaching methods	Notes
1			
Bibliography			
8.2. Applications (Seminars, Laboratory, Projects)		Teaching methods	Notes
1			
Bibliography For the diploma thesis preparation, the references are those recommended by the supervisor, as well as those obtained by studying the bibliography. For fundamental and specific knowledge assessment, the bibliography is identical to the minimal bibliography for the each of the undergraduate courses			

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

Since this topic is important for the development of a quality diploma, its content aligns the research/ design/ development topics at the European and worldwide level. The content of the course has been discussed with key actors in this area (from both the academic and industry environment).

10. Evaluation

Activity type	10.1	Assessment criteria	10.2	Assessment methods	10.3	Weight in the final grade
Course						
Applications		diploma thesis		diploma thesis		100%
10.4 Minimum standard of performance						
diploma thesis						

Course responsible
Diploma project supervisor

Head of department
Prof.dr.eng. Rodica Potolea

SYLLABUS

1. Data about the program of study

1.1	Institution	The Technical University of Cluj-Napoca
1.2	Faculty	Automation and Computer Science
1.3	Department	Computer Science
1.4	Field of study	Computer Science and Information Technology
1.5	Cycle of study	Bachelor of Science
1.6	Program of study/Qualification	Computer Science/ Engineer
1.7	Form of education	Full time
1.8	Subject code	58.

2. Data about the subject

2.1	Subject name					Defense of Diploma Thesis					
2.2	Subject area					Computer Science and Information Technology					
2.3	Course responsible/lecturer					Diploma project supervisor					
2.4	Teachers in charge of applications					As decided by the supervisor					
2.5	Year of study	IV	2.6	Semester	8	2.7	Assessment	exam	2.8	Subject category	DS/OB

3. Estimated total time

Sem.	Subject name	Lecture	Applications			Lecture	Applications			Individual study	TOTAL	Credit
		[hours / week.]				[hours / semester]						
			S	L	P		S	L	P			
8	**Defense of Diploma Thesis**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**10**

3.1	Number of hours per week		3.2	of which, course		3.3	applications	
3.4	Total hours in the teaching plan		3.5	of which, course		3.6	applications	

Individual study	Hours
Manual, lecture material and notes, bibliography	
Supplementary study in the library, online and in the field	
Preparation for seminars/laboratory works, homework, reports, portfolios, essays	
Tutoring	
Exams and tests	
Other activities	

3.7	Total hours of individual study	
3.8	Total hours per semester	
3.9	Number of credit points	10

4. Pre-requisites (where appropriate)

4.1	Curriculum	Graduating all previous disciplines from the curricula
4.2	Competence	

5. Requirements (where appropriate)

5.1	For the course	
5.2	For the applications	

6. Specific competences

Professional competences	Graduates will have the following specific skills: • modeling and designing software and hardware sub-systems, making the best decisions regarding the costs-results trade-off concerning the design decisions • implementing a hardware or software system • analyzing the way a computing system meets the criteria for which it was designed and proposing improvements and future developments • demonstrating the knowledge and understanding of important concepts, principles and theories of computer science and engineering

	• identifying and analyzing specific problems and elaborating strategies for solving them • assuring the quality of products and services in the field of information technology using the information technology tools
Cross competences	N/A

7. Discipline objectives (as results from the *key competences gained*)

7.1	General objective	Defense of Diploma Thesis
7.2	Specific objectives	

8. Contents

8.1. Lecture (syllabus)		Teaching methods	Notes
1			
Bibliography			
8.2. Applications (Seminars, Laboratory, Projects)		Teaching methods	Notes
1	• study of the bibliography in order to see how actual and necessary the project is • comparative analysis of the existing products and systems • comparative analysis of the potential methodologies and/or technologies • preparation of the project specifications • implementation and deployment of the hardware or software system • product testing and validation • product documenting • assessment of results, possible further developments, original aspects, advantages and limits of solution		
Bibliography For the diploma thesis preparation, the references are those recommended by the supervisor, as well as those obtained by studying the bibliography. For fundamental and specific knowledge assessment, the bibliography is identical to the minimal bibliography for the each of the undergraduate courses.			

9. Bridging course contents with the expectations of the representatives of the community, professional associations and employers in the field

10. Evaluation

Activity type	10.1	Assessment criteria	10.2	Assessment methods	10.3	Weight in the final grade
Course						
Applications				Separate marks - for the diploma presentation and defending (P) - for the assessment of fundamental and specific knowledge (K)		100%
10.4 Minimum standard of performance						
Exam average mark: M = (P + K) / 2 Condition to get the credits: P ≥ 5,00 ; K ≥ 5,00; M ≥ 6,00						

Course responsible
Diploma project supervisor

Head of department
Prof.dr.eng. Rodica Potolea