SYLLABUS

1. Data about the program of study

1.1 Institution	The Technical University of Cluj-Napoca
1.2 Faculty	Faculty of Automation and Computer Science
1.3 Department	Computer Science
1.4 Field of study	Computer Science and Information Technology
1.5 Cycle of study	Masters
1.6 Program of study / Qualification	Artificial Intelligence and Vision / Masters
1.7 Form of education	Full time

2. Data about the subject

2.1 Subject name	***Interactive Systems***			Subject code	**8.1**
2.2 Course responsible / lecturer	Prof. dr. eng. Dorian Gorgan - dorian.gorgan@cs.utcluj.ro				
2.3 Teachers in charge of seminars / Laboratory / project	Prof. dr. eng. Dorian Gorgan - dorian.gorgan@cs.utcluj.ro				
2.4 Year of study	I	2.5 Semester	2	2.6 Type of assessment (E - exam, C - colloquium, V – verification)	E
2.7 Subject category	Formative category: DF – fundamental, DS – speciality, DC – complementary				DF
	Optionality: DOB – mandatory, DOP – optional (alternative), DFA – optional (free choice)				DOP

3. Estimated total time

3.1 Number of hours per week	3	of which	Course	2	Seminar	1	Laboratory	-	Project	-
3.2 Total hours per semester	42	of which	Course	28	Seminar	14	Laboratory	-	Project	-

3.3 Individual study:	
(a) Manual, lecture material and notes, bibliography	20
(b) Supplementary study in the library, online and in the field	10
(c) Preparation for seminars/laboratory works, homework, reports, portfolios, essays	21
(d) Tutoring	4
(e) Exams and tests	3
(f) Other activities	-

3.4 Total hours of individual study (summ (3.3(a)…3.3(f)))	58
3.5 Total hours per semester (3.2+3.4)	100
3.6 Number of credit points	4

4. Pre-requisites (where appropriate)

4.1 Curriculum	Object oriented programming language (e.g., C++, C#)
4.2 Competence	Software application development methodology

5. Requirements (where appropriate)

5.1 For the course	Projector, Computer, Blackboard (F2F) Teaching materials will be available on MS Teams platform
5.2 For the seminar	Projector, Computer, Blackboard (F2F) Teaching materials will be available on MS Teams platform

6. Specific competences

6.1 Professional competences	• define technical requirements • analysis specifications • identifies client needs • defines system architectures • integrate system components • design process • deliver visual presentation of data • develop creative ideas • deliver visual presentation of data • creatively use digital technologies • use data processing techniques • manage system testing • align software with system architectures • create data models in the context of distributed systems • create algorithms and specific programs • designs and implements distributed resources • generates technical documentation • integrate design enterprise architecture • design information system • utilise machine learning • ensures conformity to standards • defines and implements instruments for diagnosis • conduct scholarly research • develop computer vision system • apply statistical analysis techniques • manage data collection systems
6.2 Cross competences	The graduate: • develop an analytical approach • taking a proactive approach • developing strategies to solve problems • being open minded • coordinate engineering teams

7. Expected Learning Outcomes (as results from the *key competences gained*)

Knowledge	The student has knowledge of: • algorithms • artificial neural networks • business process modelling • computer programming (Python) • computer simulation • data mining • data models • data science • digital data processing • information architecture • information categorization • information extraction • information structure • principles of artificial intelligence • systems development life cycle • task algorithmicizing • unstructured data • visual presentation techniques • machine learning (computer programming) • deep learning

	• web programming • computer programming • information structure • model-based systems engineering
Skills	The student is able to: • analyze big data • analyze business requirements • create data sets • creatively use digital technologies • define technical requirements • deliver visual presentations of data • design processes • develop creative ideas • use data processing techniques • apply systemic design thinking • build predictive models • build recommender systems • design application interfaces • identify processes for re-engineering • utilize machine learningacquire system components • align software with system architectures • analyze business requirements • create data models • define technical requirements • integrate system components • manage system testing • use an application-specific interface • use markup language
Responsibilities and autonomy	The student has the ability to work independently in order to: • develop an analytical approach • take a proactive approach • develop strategies to solve problems • be open-minded • coordinate engineering teams

8. Discipline objective (as results from the *key competences gained*)

8.1 General objective	The main objective of the discipline is the development of interactive applications by learning the techniques of analysis, specification, design, implementation and evaluation of the components that ensure interaction with the user. The concepts and techniques of software engineering specific to user-oriented methodologies are highlighted.
8.2 Specific objective	To achieve these general objectives, students will learn to: • Design the architecture of interactive software systems. • Use software tools for the development of interactive applications. • Carry out a bibliographic and experimental research activity, the results of which are written in a scientific paper. • Produce a scientific synthesis and analysis and, to support an oral presentation of a scientific topic. • Carry out a project according to the methodology for the development and evaluation of user-oriented interactive applications. • Work individually or in a team.

9. Contents

9.1. Lectures	Hours	Teaching methods	Notes
Introduction. History	2		

Concepts of user interface development	2	Multimedia teaching techniques are used in the course. The course is interactive with demonstrations, to exemplify the methodology of developing interactive applications.	Consultations are planned during the semester and before each exam.
Inputs and outputs communication concepts	2		
User-oriented design	2		
User interface design methodology	2		
Usability in graphical user interfaces	2		
Defining user requirements. Task description and analysis	2		
User interface prototyping. User interface evaluation	2		
Techniques and styles of interaction	2		
Conceptual architecture of a computer game	2		
Game engines. The architecture of a game engine	2		
Computer games development methodology	2		
Strategies in computer games	2		
The components of a strategy. Strategic options	2		

Bibliography (minimum bibliography of the discipline containing at least one bibliographic reference work of the discipline, which is available to students in a corresponding number of copies)

In the UTC-N library

1. B. Shneiderman, C. Plaisant, M.S. Cohen, S.M. Jacobs, Designing the User Interface. Strategies for Effective Human Computer Interaction, Addison-Wesley (5th Edition).
2. A. Watt, F. Policarpo, 3D Games: Real-time Rendering and Software Technology, Addison-Wesley, 2000.

In virtual library

1. Interactive Sistems Course resources, https://moodle.cs.utcluj.ro/

9.2. Applications - Seminars/Laboratory/Project	Hours	Teaching methods	Notes
Computer game development methodology.	1	Case studies on topics in the field of interactive applications, examples using software tools and specialized technologies, presentations on the board, additional explanations, discussions.	Each student develops a project based on initial requirements.
Computer game project proposal.	1		
Game analysis and specification. Task description and analysis. Low fidelity prototyping. Game scenarios.	1		
Computer game design. Scene of objects. Game strategy. Interaction techniques. Artificial intelligence techniques.	1		
Technology and Implementation Considerations. Unity technology. Game implementation using Unity technology.	1		
Game rating. Evaluation of functional correctness. Usability evaluation. Heuristic evaluation. Evaluators' reports.	1		
Development of the final version of the game. Conclusions and final presentation of the project. Presentation of game execution. Presentation of the project report.	1		

Bibliography (minimum bibliography of the discipline containing at least one bibliographic reference work of the discipline, which is available to students in a corresponding number of copies)

In the UTC-N library

1. A. Watt, F. Policarpo, 3D Games: Real-time Rendering and Software Technology, Volume 1, Addison-Wesley.
2. A. Watt, F. Policarpo, 3D Games, Volume 2: Animation and Advanced Real-time Rendering, Addison-Wesley.

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

Unlike established software engineering, the course presents the user-oriented methodology widely used in the development of interactive applications. Techniques specific to this methodology based on the concept of usability, user scenarios, prototyping, metaphors, cognitive evaluation, heuristic evaluation, multimodal interaction, etc. are studied and exemplified. The content of the discipline was discussed with important actors in this field, both from academia and industry, from Romania or other countries. The discipline has been evaluated by ARACIS, along with other master's study programs.

11. Evaluation

Activity type	Assessment criteria	Assessment methods	Weight in the final grade
Course	The written exam (E) assesses the knowledge taught in the course. Course activity (CA) reflects active participation in scientific presentations and debates in the course.	Written examination. Assessment during the semester. Online tests using distance education platforms.	30% (E) 10% (CA)
Seminars	The scientific paper (L) demonstrates the ability to carry out a scientific study and the elaboration of a scientific paper or report. The project (P) demonstrates the ability to use the interactive application development methodology.	Scientific paper, Project. Project presentation onsite or online using distance education platform.	30% (L) 30% (P)
Minimum standard of performance Final mark: M = 0,3*E + 0,3*L + 0,3*P + 0.1*CA Condition for obtaining credits: M≥5; E ≥ 5; L ≥ 5; P ≥ 5; CA ≥ 5.			

Date of filling in: 29.04.2026	**Responsible**	**Title First name Surname name**	**Signature**
	Course	Prof.dr.eng. Dorian GORGAN	
	Applications	Prof.dr.eng. Dorian GORGAN	

Date of approval in the department	Head of department, Prof.dr.eng. Rodica Potolea
Date of approval in the Faculty Council	Dean, Prof.dr.eng. Vlad Mureșan