COURSE DESCRIPTION CARD - SYLLABUS

Course name
Specification and description languages

Course

Field of study
Electronics and Telecommunications
Area of study (specialization)
Computer networks and Internet technologies
Level of study
Second-cycle studies
Form of study
full-time

Year/Semester
1/2
Profile of study
general academic
Course offered in
Polish
Requirements
elective

Number of hours

Lecture
30
Laboratory classes
15
Other (e.g. online)
Tutorials
Projects/seminars
15

Number of credit points

4

Lecturers

Responsible for the course/lecturer:
prof. dr hab. inż. Grzegorz Danilewicz,
grzegorz.danilewicz@put.poznan.pl

Responsible for the course/lecturer:

Prerequisites

Student knows the basics of telecommunications systems and has knowledge of programming. Can obtain information from literature, databases and other sources in Polish or English, interpret them, draw conclusions and justify opinions. Can communicate in Polish or English in a professional environment.

Student can prepare, in Polish or English, a well-documented study of problems in the field of electronics and telecommunications. Can prepare an oral presentation in Polish or English on specific

issues in the field of electronics and telecommunications. Student can educate himself. Knows the limitations of his own knowledge and skills, understands the need for further training.

Course objective

To familiarize students with the specificity of software used in telecommunications systems and with formal methods of designing telecommunications software. Using formal languages for the specification, description and design of systems.

Course-related learning outcomes

Knowledge

1. Has knowledge of the specificity of real-time systems software. Has knowledge of the specificity of the software of communicating systems. Has knowledge of the specificity of the software for small- and large-scale telecommunications equipment.

2. Has knowledge of modeling real-time systems using the concept of finite-state machines. Has knowledge of modeling the communicating systems with a finite-state machines.

3. Has basic knowledge in the field of software engineering that allows the implementation of projects related to planning, specification, description, implementation and software testing.

Skills

1. Can collect and analyze technical information needed to create software for simple communicating systems, can present these issues in the form of text studies and presentations (in Polish or English), can argue in the discussion on the presented issues.

2. Is able to use data bases collecting standards for telecommunications, knowing the importance of standardization, can take into account the limitations contained in the standards when designing a software for telecommunications devices.

3. Can practically carry out selected tasks of creating software for communicating systems.

Social competences

1. Understands the importance of communication for the development of individuals and societies, understands the evolutionary development of telecommunications networks and systems, takes into account the growing needs of users in the development of telecommunications systems.

2. Is aware of the need for a professional approach to solved technical problems and taking responsibility for the proposed technical solutions.

3. Knows the limitations of his own knowledge and skills, understands the need for further training.

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

In the laboratory a grade is based on: the basis of preliminary questions, answers to questions about the material from the previous laboratory, the basis of written reports of laboratory and the tests.

The final grade is the result of component grades, with each component grade being positive. The rating scale from 2 (insufficient - negative) to 5 (very good) is used for component grades and for the final grade.

Project: preparing a text, preparing at least one presentation, delivering a paper on a given topic in the field of description and specification languages, initiating, participating and concluding the discussion. Projects can be realized as individual or 2-person team.

The final rating for the project is the result of the grades for the individual components:

1. Class attendance

2. Activity in class, involvement in discussions, the ability to defend one's position

3. Quality of a presentation and text

4. Ability to deliver a speech

5. Timely execution of the tasks

The final grade is a product of the component rates, where each of the component scores must be positive. The rating scale from 2 (insufficient - negative) to 5 (very good) is used for a final project grade as well as for component rates.

In respect of lectures a grade is based on test and/or oral examination. Test is composed of 10-15 closed and open questions. A positive test grade is issued when the number of points exceeds 50%. The rating scale from 2 (insufficient - negative) to 5 (very good) is used for an exam grade. It is allowed to lower the threshold by a maximum of 10%.

Programme content

Lectures:

The specificity of telecommunications software. Introduction to formal languages and comparison with natural languages. Formal description, formal specification. Protocol engineering, software engineering, telecommunications software engineering. Finite state machines, definitions of states and events. Formal representations of FSM. Extended FSM machines. Telecommunications systems as EFSM. Communicating EFSM. MSC Message Sequence Charts. SDL description and specification language. Abstract ASN.1 notation. Fundamentals of software engineering. Object-oriented modeling. Telecommunications software testing. Standardization issues.

Laboratories:

Creating a formal specification of a communicating system, writing software that uses communication between computers, using the concept of a finite-state machine.

Teaching methods

Lectures with the use of a projector/whiteboard, conversation lecture, experiment, case study, talk, participation in the discussion, steering the discussion

Bibliography

Basic

1. Międzynarodowy Związek Telekomunikacyjny (ITU-T) Specification and Description Language (SDL): Overview of SDL-2010, Zalecenie Z.100 z późniejszymi zmianami

2. Ian Somerville: Inżynieria oprogramowania, WNT, 2003

Additional

1. Ian Somerville: Software Engineering, Pearson Education Limited, 2001

2. Grady Booch, James Rumbaugh, Ivar Jacobson: UML przewodnik użytkownika, WNT 2002

3. Miroslav Popovic: Communication Protocol Engineering, Taylor & Francis, 2006

4. Stanisław Szejko, red.: Metody wytwarzania oprogramowani, Mikom, 2002

Breakdown of average student's workload

	Hours	ECTS
Total workload	100	4,0
Classes requiring direct contact with the teacher	70	3,0
Student's own work (literature studies, preparation for laboratory classes, preparation for tests/exam, project preparation) [1]	30	1,0

[1] delete or add other activities as appropriate