



COURSE DESCRIPTION CARD - SYLLABUS

Course name

IT in Administration [S2Inf1E-IO>ITADM]

Course

Field of study

Computing

Year/Semester

2/3

Area of study (specialization)

Software Engineering

Profile of study

general academic

Level of study

second-cycle

Course offered in

English

Form of study

full-time

Requirements

compulsory

Number of hours

Lecture

20

Laboratory classes

30

Other

0

Tutorials

0

Projects/seminars

0

Number of credit points

3,00

Coordinators

prof. dr hab. inż. Jerzy Nawrocki
jerzy.nawrocki@put.poznan.pl

mgr inż. Piotr Miklosik
piotr.miklosik@put.poznan.pl

Lecturers

Prerequisites

-

Course objective

1. To give students the knowledge about IT systems used in administration, with particular emphasis on systems used in higher education. 2. to provide students with knowledge about the creation of IT systems created to meet the specific needs and intended for a specific group of users. 3. to develop students' skills in software engineering with a special emphasis on system architecture development. 4. to develop students' skills in solving problems connected with designing, implementing and using IT systems in administration. 5. to develop students' skills in organizing work related to the development and implementation of IT systems.

Course-related learning outcomes

Knowledge:

he/she has detailed theoretical knowledge in selected areas of computer science.
he/she has a basic knowledge of new trends and solutions concerning IT systems used in administration.
he/she has knowledge of software engineering concerning computer systems architecture, human-computer interaction and quality management.
he/she knows basic methods, techniques and tools used to solve engineering tasks in IT systems used in administration.
he/she has knowledge of the life cycle of information systems.

Skills:

he/she is able to experimentally assess the usefulness and applicability of new solutions (methods and tools) and new IT systems in administration.
he/she can acquire, combine, interpret and evaluate information from literature, databases and other sources (in their native language and English); he/she can draw conclusions and form opinions on the basis of them.
he/she can apply experimental methods to formulate and solve engineering tasks.
he/she can analyze existing solutions in order to introduce innovations.
he/she can combine knowledge from different areas of computer science (and other disciplines if necessary) to formulate and solve engineering problems and can use approaches that integrate non-technical aspects.
he/she can formulate and test hypotheses on engineering problems, and is able to solve complex computer tasks, including non-routine ones.

Social competences:

he/she knows examples and understands the reasons for failures of IT systems that have led to large financial and social losses, or have resulted in the endangerment of life or health.
he/she understands that skills and knowledge in IT are rapidly becoming obsolete.

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Learning outcomes presented above are verified as follows:

Formative assessment:

Based on the answers given regarding the performance of the laboratory tasks.

Summative assessment:

On the basis of the report or presentation prepared by students on the computerization of the selected branch of administration together with the risk analysis.

Programme content

During lectures students get acquainted with IT projects supporting administration conducted at the University, as well as with other IT systems supporting administration in a broad sense. They also learn about problems and challenges (technical and non-technical) resulting from conducted implementations.

During laboratory classes, students solve problems typical of IT systems used in administration and prepare proposals for their own systems to support administration, along with a proposal to solve problems associated with their implementation.

The following topics are discussed in detail:

- Architecture of many systems used in administration
- Common problems encountered during implementation
- Interaction with end users
- Maintaining IT systems
- Risk analysis
- Reasons for system failures (technical and non-technical)
- Quality in IT systems
- Business aspects connected with designing and implementing IT systems

Course topics

During lectures students get acquainted with IT projects supporting administration conducted at the University, as well as with other IT systems supporting administration in a broad sense. They also learn

about problems and challenges (technical and non-technical) resulting from conducted implementations.

During laboratory classes, students solve problems typical of IT systems used in administration and prepare proposals for their own systems to support administration, along with a proposal to solve problems associated with their implementation.

The following topics are discussed in detail:

- Architecture of many systems used in administration
- Common problems encountered during implementation
- Interaction with end users
- Maintaining IT systems
- Risk analysis
- Reasons for system failures (technical and non-technical)
- Quality in IT systems
- Business aspects connected with designing and implementing IT systems

Teaching methods

Lecture: multimedia presentation, discussion, showcase.

Laboratory: problem solving, discussion, teamwork, demonstration.

Bibliography

Basic

1. P. Kruchten: Architectural Blueprints—The 4+1 View Model of Software Architecture, November 1995.

Additional

1. Software Requirements 2nd edition, K. E. Wiegers, Microsoft Press, Redmond, WA, USA, 2003

Breakdown of average student's workload

	Hours	ECTS
Total workload	75	3,00
Classes requiring direct contact with the teacher	50	2,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	25	1,00