COURSE DESCRIPTION CARD - SYLLABUS

Course name
Content Management Systems [N2Inf1-ZTI>SZT]

Course

Field of study
Computing

Year/Semester
1/1

Area of study (specialization)
Advanced Internet Technologies

Profile of study
general academic

Level of study
second-cycle

Course offered in
Polish

Form of study
part-time

Requirements
compulsory

Number of hours

Lecture
12

Laboratory classes
16

Other
0

Tutorials
0

Projects/seminars
0

Number of credit points

3,00

Coordinators

dr inż. Marcin Borowski
marcin.borowski@put.poznan.pl

Lecturers

dr inż. Marcin Borowski
marcin.borowski@put.poznan.pl

dr inż. Michał Apolinarski
michal.apolinarski@put.poznan.pl

Prerequisites

A student starting this subject should have basic knowledge of programming structured and object-oriented programming, programming using the MVC scheme, basic knowledge of web technologies (HTML, CSS, JS), and basic knowledge of database design. He/she should have the ability to solve basic problems related to the information systems design process and the ability to obtain information from indicated sources. He or she should also understand the necessity to broaden his or her competences / have the readiness to cooperate within a team. Moreover, in terms of social competence, the student must present such attitudes as honesty, responsibility, perseverance, cognitive curiosity, creativity, personal culture, respect for other people.

Course objective

1. to provide students with knowledge concerning the design of content management systems, information flow in such systems, technologies used in the construction of content management systems, in terms of approaches to the design of CMS (including for mobile devices). 2) Developing in students the ability to solve problems related to the design of Internet applications, content management systems, using open source solutions, frameworks and libraries supporting the construction of such solutions. 3. shaping in students the ability to work in a team and to solve problems independently.

Course-related learning outcomes

Knowledge:
Student:
• has an organized, theoretically founded knowledge of network technologies and internet applications related to the construction of CMS
• has detailed knowledge related to selected issues in the field of IT used in the construction of content management systems (XML, REST, data repositories, processes in CMS)
• has knowledge of development trends in IT and in selected related disciplines - technologies used in CMS construction
• has knowledge about the life cycle of IT systems (content managment systems), and the life cycle of data processed in IT

Skills:
Student:
• can, when formulating and solving engineering tasks, integrate knowledge from various areas of computer science (and if necessary knowledge from other scientific disciplines) as well as knowledge from the area of operation of a given CMS and apply a systemic approach, also taking into account non- technical aspects
• can assess the usefulness and the possibility of using new technological achievements (methods, tools, libraries, frameworks, services) and new IT products
• can use analytical, simulation and experimental methods (such as: estimating the number of requests to CMS, load the server with SQL queries) to formulate, solve engineering tasks and simple research problems , can correctly design and implement efficient mechanisms
• can make a critical analysis of the existing technical solutions used by the enterprise and propose their improvements (improvements) in the form of CMS components
• can, in accordance with the given specification, taking into account non-technical aspects and requirements - design a complex CMS and implement this project - at least in part - using appropriate methods, techniques and tools, including adapting existing or developing new tools for this purpose

Social competences:
Student:
• understands that in computer science knowledge and skills very quickly become obsolete, in particular internet and mobile technologies
• understands the need to use the latest technological achievements and knows examples and understands the causes of malfunctioning of CMS systems, which may lead to serious financial, image or social losses

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Formative assessment
a. lecture - on the basis of activity during the interactive parts of the lectures;
b. laboratory - on the basis of the evaluation of the current progress of the tasks;
Summative assessment
a. lecture
- Evaluation of the acquired knowledge and skills demonstrated by an exam - a test conducted with the use of an Internet application; questions of general and problematic nature (the student using an Internet application; questions of a general and problematic nature (a student may use any didactic materials; a choice test with approx. 30 questions; a mark in accordance with the criteria as follows:

> 50% - 3.0
> 60% - 3.5
> 70% - 4.0
> 80% - 4.5
> 90% - 5.0
- discussion of examination results
b. laboratory
- verification of the assumed learning outcomes realised by:
- assessment and defence by students of prepared assignments - 5 projects;
When calculating the final grade, the student may obtain a higher grade for:
- discussion of additional aspects of the presented issues, not presented in class;
- the use of skills and knowledge from outside the curriculum to solve implemented tasks;
- assistance in improving teaching materials related to the subject;

Programme content

Lecture:
The lecture programme covers the following topics:
Basic information about CMS. Division of content management systems. Discussion of content management system components, components and processing methods. Basics of XML, DTD, XML Schema, Relax NG. Business processes - a reminder from previous years' subjects. Information on process management systems. Discussion of issues related to work processes. Discussion of issues related to the basic elements and mechanisms of a CMS.
Laboratory:
Laboratory classes are conducted in the form of fifteen 2-hour exercises, taking place in the laboratory. The exercises are carried out independently by the students. The laboratory programme covers the following topics:
Review and analysis of popular/selected CMS systems. Design and implementation of a dedicated content management system (including multilingual) based on open-source and/or proprietary solutions. Development of system design documentation including: functional and non-functional requirements, UML diagrams, interface prototypes (UX mockup), database design, test results (performance and OWASP security audit). Consideration of the latest technologies and trends in the design.

Course topics

Lecture:
- Basic information about content management systems (CMS).
- Division of content management systems.
- Content management systems components, components and processing methods.
- Basics of XML, DTD, XML Schema, Relax NG.
- Business processes (a reminder from earlier).
- Process management systems.
- Work processes.
- Basic elements and mechanisms of CMS.
Lab:
- 15 two-hour laboratory exercises.
- Exercises carried out independently by students.
Exercise topics:
- Review and analysis of popular CMS systems.
- Design and implementation of a dedicated content management system, including multilingual, using open-source or in-house solutions.
- Development of project documentation:
 - Functional and non-functional requirements.
 - UML diagrams.
 - Interface prototypes (UX mockup).
 - Database design.
 - Test results (performance and OWASP security audit).
- Consideration of the latest technologies and trends in the design.

Teaching methods

1. lecture: multimedia presentation, presentation illustrated with examples given on the blackboard, solving tasks, multimedia show, demonstration.
2. laboratory exercises: practical exercises, discussion, teamwork, multimedia show, case study, demonstration.

Bibliography

1. Document engineering, Robert J. Glushko, Tim McGrath, MIT Press, Cambridge, 2005
2. Technical documentation of these tools available on the internet
3. https://www.w3schools.com/ (selected issues)
4. Django documentation (http://djangoproject.com).
5. the Python documentation (http://python.org)

Breakdown of average student's workload

	Hours	ECTS
Total workload	75	3,00
Classes requiring direct contact with the teacher	28	1,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	47	2,00