COURSE DESCRIPTION CARD - SYLLABUS

Course name

Internet Systems

Course

Field of study	Year/Semester
Computing	1/1
Area of study (specialization)	Profile of study
Intelligent Information Technologies	general academic
Level of study	Course offered in
Second-cycle studies	Polish
Form of study	Requirements
full-time	compulsory

Number of hours

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

Number of credit points

5

Lecturers

Responsible for the course/lecturer:	Responsible for the course/lecturer:
Tomasz Pawlak, PhD	

Prerequisites

The student starting this course should have a basic knowledge of the development of computer systems, programming and the architecture of computer systems. In particular, the student is required to be familiar with the Java programming language.

Course objective

1. Provide students with basic knowledge about the use of programming tools in the field of web systems development.
2. Developing students' problem-solving skills related to analysis and design of Internet systems as part of technology used in business.
3. Discussing issues related to the security and performance of online systems used in business.
4. Teaching students the skills of teamwork as part of laboratory exercises during the implementation of technical projects in groups.

Course-related learning outcomes

Knowledge

1. The student has structured, theoretically founded general knowledge in the field of internet systems

architecture and technologies for designing and implementing internet systems - [K2st_W1]
2. The student has a structured and theoretically founded general knowledge related to key issues in the field of online systems - [K2st_W2]
3. The student has a theoretically well-founded knowledge related to issues in the field of internet systems, such as: designing, creating and testing an internet system - [K2st_W3]
4. The student has knowledge about development trends and the most important new IT achievements in the field of internet systems - [K2st_W4]
5. The student has a basic knowledge of the life cycle of online systems - [K2st_W5]

Skills
1. The student is able to formulate and test hypotheses related to engineering problems - [K2st_U4]
2. The student is able - when formulating and solving engineering tasks - to integrate knowledge from various areas of computer science (and if necessary also knowledge from other scientific disciplines) and apply a systemic approach, also taking into account non-technical aspects - [K2st_U5]
3. The student can assess the suitability and the possibility of using new IT solutions in the field of web systems - [K2st_U6]
4. The student is able to assess the usefulness of methods and tools for solving the engineering task, consisting in the construction of an internet system, including the technological limitations - [K2st_U9]
5. The student is able to choose a programming language suitable for a given programming task - [K2st_U11]

Social competences
1. The student understands that in informatics, knowledge and skills quickly become obsolete - [K2st_K1]
2. The student understands the importance of using the latest knowledge in the field of computer science in solving research and proctological problems - [K2st_K2]

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:
Forming rating:
a) in the scope of lectures:
- based on the answers to questions about the material discussed in lectures,
b) in the scope of laboratories:
- on the basis of an assessment of the current progress of laboratory tasks.
Summary rating:
a) in the scope of lectures, verification of the assumed learning outcomes is accomplished by:
- assessment of knowledge and skills demonstrated on written colloquium consisting of:
- a set of closed questions, each of which can be answered with one correct answer from four possible ones. One point is obtained for each correct answer, and 1/3 of the point is taken for incorrect answer.
- a set of open questions, for which you can get from 2 to 4 points for each.
To get a grade 3.0, a minimum of 51% points should be obtained, 3.5 - 61%, 4.0 - 71%, 4.5 - 81%, 5.0 - 91%.
b) in the scope of laboratories, verification of the assumed learning outcomes is carried out by:

- assessment of skills related to the implementation of laboratory exercises including programming,
- evaluation of the report prepared partly during the course and partly after its completion; this assessment also includes the ability to work in a team.

Obtaining additional points for activity during classes, especially for:
- discuss additional aspects of the issue,
- effectiveness of using the acquired knowledge while solving a given problem,
- ability to cooperate in a team practically performing a detailed task in a laboratory,
- comments related to the improvement of didactic materials,
- presentations of the self-presentation related to the subject of the classes.

Programme content

The lecture program includes the following topics:
- Communication in internet systems. The lecture introduces issues related to HTTP, XML-RPC protocols and Web-services. The lecture introduces to XML-RPC protocol for data exchange using XML frames. The next part of the lecture is devoted to the introduction of Web Services technology using the HTTP protocol. The pattern of REST network services will also be discussed.
- Technologies supported by the browser: HTML, CSS. Language functionality and less popular features influencing the ergonomics of using the web application will be discussed. The extensions available in HTML 5 and CSS 3 will be presented.
- Technologies supported by the browser: JavaScript, DOM, XSLT (XPath). Advanced methods of using the language will be presented, eg building lambda expressions, pseudo-objectivity and tools supporting programming in JavaScript. The DOM model of HTML and XML documents will be presented as well as the associated API.
- Ajax and JSON mechanism. The operation of the mechanism of access to data from other websites (Cross-origin resource sharing) will be discussed. The lecture will discuss the Ajax mechanism in detail. New solutions for communication with the server will be introduced as part of the HTML 5 project. Next, the JSON data transfer format and the jQuery library will be presented to facilitate manipulation of the DOM tree and execution of AJAX queries.
- Dynamic generation of websites and web resources: Servlets and JSP. The lecture introduces to the mechanisms of dynamic website generation and the structure of the Internet application in JEE technology. The Servlet technology in the WWW containers in JEE technology will be discussed. Next, the application of JSP templates for generating HTML pages and other shared resources will be shown.
- Data persistence in the web application on the example of JDBC and JPA. The lecture presents the methods of data storage and retrieval using Java Database Connectivity (JDBC). A JPA interface will be discussed for storing Java objects in the database. Discussion of relational databases and NoSQL databases on the example of MongoDB, Redis and Apache Cassandra systems.
- MVC pattern. Review of environments compatible with the MVC pattern and its derivatives.
- Different ways to create web applications on Dart and MEAN stack examples: MongoDB + Express + AngularJS + Node.js. Two lectures include a discussion of popular programming environments that allow you to create applications that are compatible with the MVC model. The environments present various paradigms used in programming web applications.
- Security, performance and correctness of Internet applications. Two lectures introduce issues related

to the creation of secure and efficient internet applications. Basic threats that may be encountered while using Internet applications will be discussed. Methods of measuring the web application's performance and factors affecting it will be presented.

Laboratory classes are conducted in the form of fifteen two-hour exercises held in the laboratory, preceded by an instructional session at the beginning of the semester. Exercises are carried out independently. Exercises are carried out on the basis of didactic materials provided by the teacher. The laboratory program includes the following issues:

- Implementation of introductory exercises in HTTP technology - 8 hours
- Representational State Transfer - 4 hours
- NoSQL databases - 4 hours
- Hypertext Markup Language - 2 hours
- Cascading Style Sheets - 2 hours.
- JavaScript - 4 hours.
- Discussion on difficulties in the realization of the above mentioned issues and consultations - 12 hours.

Teaching methods

1. Lecture: multimedia presentation.
2. Laboratory exercises: solving tasks, practical exercises, team work, demonstration.

Bibliography

Basic

1. Krzysztof Rychlicki-Kicior: Java EE 6 : programowanie aplikacji www, Helion, 2010
2. Steven Sanderson, Adam Freeman: ASP.NET MVC 3 framework : zaawansowane programowanie, Helion, 2012
3. Semmy Peruwal: Learning Web App Development, O'Reilly Media, 2014
4. Ethan Brown: Web Development with Node and Express, O'Reilly Media, 2014

Additional

1. Technical documentation of WWW standards: http://www.w3.org
2. Technical documentation of Rails and Dart: http://rubyonrails.org/, https://www.dartlang.org/
3. Technical documentation of J2EE standard: http://www.jcp.org

Breakdown of average student's workload

	Hours	ECTS
Total workload	125	5,0
Classes requiring direct contact with the teacher	60	2,5
Student's own work (literature studies, preparation for laboratory classes, homework, preparation for tests) [1]	65	2,5

[1] delete or add other activities as appropriate