



## COURSE DESCRIPTION CARD - SYLLABUS

Course name

Programming Internet Applications [N2Inf1-ZTI>PAI]

### 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

16

Laboratory classes

16

Other

0

Tutorials

0

Projects/seminars

0

### Number of credit points

4,00

### Coordinators

dr inż. Marcin Borowski

marcin.borowski@put.poznan.pl

### Lecturers

dr inż. Marcin Borowski

marcin.borowski@put.poznan.pl

dr inż. Andrzej Urbański

andrzej.urbanski@put.poznan.pl

### Prerequisites

In terms of knowledge, the student should have basic knowledge of programming structured and object-oriented programming, basic knowledge of web technologies and basic knowledge of designing and using databases. He or she should have the ability to solve basic problems related to the process of designing information systems and the ability to obtain information from indicated sources. In terms of social competences, the student should demonstrate such attitudes as honesty, responsibility, perseverance, cognitive curiosity, creativity, personal culture, respect for other people.

### Course objective

1. Provide students with basic knowledge on the design of web and mobile applications, the flow of information in such applications, technologies used in their construction (including applications for mobile devices). 2. Developing students' skills in solving problems related to the design of web and mobile applications, the use of Open Source solutions, frameworks, and libraries supporting the construction of such solutions. 3. Shaping students' teamwork skills and independence in solving problems.

## Course-related learning outcomes

### Knowledge:

#### Student:

- has an orderly, theoretically founded knowledge of network technologies and internet applications related to the construction of internet applications
- has detailed knowledge related to selected issues in the field of computer science used in building web applications
- has knowledge of development trends in IT and in selected related disciplines - technologies used to build internet applications
- knows the life cycle of web applications, the life cycle of data processed in web applications
- knows advanced methods and techniques used in the implementation of engineering tasks related to building web applications

### 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 in the area of operation of a given internet application and apply a system approach, also taking into account non-technical aspects
- can obtain information on Internet applications from literature, databases, and other sources (in Polish and English), integrate them and interpret and evaluate them, draw conclusions and formulate opinions
- can assess the usefulness and the possibility of using new technological achievements (methods, tools, libraries, frameworks, services) and new IT products
- can determine the directions of further learning and gain new knowledge about web applications and pass it on to others

### Social competences:

#### Student:

- understands the need to use the latest technology achievements and knows examples and understands the causes of malfunctioning applications that may lead to serious financial, image, or social losses
- understands that in computer science, knowledge and skills very quickly become obsolete, especially internet and mobile technologies
- can set priorities for the implementation of tasks defined by himself or others, in particular during the implementation of IT projects

## 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

- a. Lecture: based on activity during the interactive parts of the lectures;
- b. laboratory: based on the assessment of the current progress in the implementation of tasks;

### Summative assessment

#### a. Lecture:

- assessment of the acquired knowledge and skills shown in the exam - a test conducted using an internet application; general and problem-related questions (the student can use any teaching materials; a selection test with about 30 questions; pass according to the following criteria of the obtained result:

> 50% - 3.0

> 60% - 3.5

> 70% - 4.0

> 80% - 4.5

> 90% - 5.0

- discussion of the exam results

#### b. Laboratory:

- verification of the assumed learning outcomes realized by
  - students' assessment and defense of the prepared tasks - 5 small projects;
- When assigning the final grade, the student may obtain an increase in grade for:
- discussing additional aspects of the presented issues, not presented during classes;
  - using skills and knowledge from outside the study program to solve the tasks performed;
  - help in improving teaching materials related to the subject;

## Programme content

Lecture:

The lecture programme covers the following topics:

Structure and syntax of HTML5, CSS3, JavaScript. Basics of node.js, express.js and some current frontend tools.

Laboratory:

Laboratory classes are conducted in the form of sixteen hour-long classes held in a computer laboratory. The first class is designed to familiarise students with rules of using the laboratory and passing the exercises.

The syllabus of the laboratory classes includes the following topics:

Creating pages in HTML5. Creating CSS3, LESS, SASS stylesheets. Basics of the language JavaScript and auxiliary libraries. Building applications using the discussed modern front-end technologies.

## Course topics

### Lecture:

- Structure and syntax of HTML5, CSS3, JavaScript.
- Basics of Node.js and Express.js.
- Current frontend tools.

### Lab:

- Creating pages in HTML5.
- Creation of CSS3 stylesheets, LESS, SASS.
- Fundamentals of JavaScript and supporting libraries.
- Building applications with modern frontend technologies.

## Teaching methods

Lecture: multimedia presentation, illustrated by examples given on the blackboard.

Laboratory exercises: multimedia presentation, illustrated by examples given on the blackboard, live coding and performance of tasks given by the instructor - practical exercises.

## Bibliography

- Bootstrap Framework – <http://getbootstrap.com>
- TailwindCSS - <https://tailwindcss.com/>
- Javascript - <https://www.javascript.com/>
- Svelte - <https://svelte.dev/>

## Breakdown of average student's workload

	Hours	ECTS
Total workload	100	4,00
Classes requiring direct contact with the teacher	34	1,50
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	66	2,50