



COURSE DESCRIPTION CARD - SYLLABUS

Course name

Creation of internet and mobile applications [S2Bioinf2>TAIM]

Course

Field of study
Bioinformatics

Year/Semester
1/2

Area of study (specialization)
–

Profile of study
general academic

Level of study
second-cycle

Course offered in
Polish

Form of study
full-time

Requirements
elective

Number of hours

Lecture
30

Laboratory classes
30

Other
0

Tutorials
0

Projects/seminars
0

Number of credit points

4,00

Coordinators

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

Lecturers

Prerequisites

In terms of knowledge, the student should have basic knowledge about structured and object-oriented programming, basic knowledge of internet technologies and basic knowledge corresponding to database designing and using. The student should have the ability to solve basic problems related to the process of designing IT systems and the ability to obtain information from the indicated sources. In terms of social competences, the student must present attitudes such as honesty, responsibility, perseverance, cognitive curiosity, creativity, personal culture, respect for other people.

Course objective

1. Provide students the basic knowledge about designing of web and mobile applications, information flow in such applications, and technologies used in their construction. 2. Developing students' skills in solving problems related to the designing of web 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:

1. A student has an orderly, founded general knowledge in terms of internet technologies and mechanisms used in internet and mobile applications.
2. A student has detailed theoretical knowledge related to selected issues in the field of computer science.
3. A student knows the basic methods, techniques and tools used to solve complex engineering tasks in a selected area of computer science and technology.

Skills:

1. A student can obtain information from literature, databases, the Internet and other sources (in Polish and English), integrate them, interpret and critically evaluate them, draw conclusions and formulate and exhaustively justify opinions, in particular use various types of technical documentation and API tools used during practical classes on building internet and mobile applications.
2. A student can use acquired mathematical knowledge to describe processes, create models and write algorithms.
3. A student can design a device, IT system or process by the given specification, using appropriate methods, techniques and tools.

Social competences:

1. Is aware of the dynamic nature of the development of information technology, in particular in the area of content management systems and web and mobile technologies, and understands the need for continuous improvement of his/her skills.
2. Is able to identify priorities in an entrepreneurial manner in carrying out tasks related to the implementation, maintenance or development of content management systems, both for him/herself and the team, taking into account the latest technological developments.
3. Understands the potential consequences of malfunctioning systems, including financial, reputational or social losses, and seeks to minimise risks through a responsible and proactive approach to project delivery.

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Formative assessment:

In the field of lectures:

- based on answers to questions about the material discussed in previous lectures,

In the field of laboratories:

- based on the assessment of the current progress of the implementation of the tasks

Summative assessment:

In the field of lectures, verification of the assumed learning outcomes is carried out by:

- assessment of the knowledge and skills are shown in the problematic test (the student may use any teaching materials) in the form of a selection test - 30 questions for a total of 60 points (2 points for the correct answer), passing from 60% + 1
- discussion of the results of the test,

In the field of laboratories, verification of the assumed learning outcomes is carried out by:

- assessment and "defence" by the student of the report on the implementation of the project,

Obtaining additional points for activity during classes, especially for:

- discuss additional aspects of the issue,
- the effectiveness of applying the acquired knowledge while solving a given problem,
- notes related to the improvement of teaching materials,
- identifying students' perceptual difficulties enabling ongoing improvement of the teaching process.

Programme content

The lecture program covers the following topics:

Structure and syntax of HTML5, CSS3. Basics of JavaScript language, the use of the jQuery library. Programming in PHP. Overview of the Yii2 framework and creating applications based on this framework. Modern frontend technologies (ReactJS, Angular, Svelte). A framework ReactNative will be presented in the field of mobile technologies.

Laboratory exercises are conducted in the form of fifteen two-hour classes taking place in the computer laboratory. The first classes are intended to familiarize students with the rules of using the laboratory and completing the exercises. Classes are carried out by two-person teams of students. The laboratory program covers the following topics:

HTML5 website development. Creating CSS3 style sheets. Using JavaScript libraries. Programming in PHP.

Building an application using the Yii2 framework. Building frontend applications in the discussed technologies.

Course topics

The course covers the design, creation and implementation of modern web and mobile applications using current web technologies. The programme focuses on languages and standards such as HTML5, CSS3, JavaScript and TypeScript, as well as ReactJS, Angular, Svelte and React Native frameworks. Students learn practical skills in building user interfaces, integrating components, working in teams, and using open source libraries and tools that support application development.

Teaching methods

1. lectures: multimedia presentation, presentation illustrated with examples given on the board, solving problems, multimedia show, demonstration.
2. laboratory exercises: practical exercises, discussion, teamwork, multimedia show, case studies, demonstration.

Bibliography

Basic:

1. HTML5 – <https://developer.mozilla.org/en-US/docs/Web/HTML>
2. CSS3 – <https://developer.mozilla.org/en-US/docs/Web/CSS>
3. Tailwindcss Framework – <https://tailwindcss.com/>
4. SvelteKit Documentation - <http://svelte.dev>
5. ReactJS – <https://pl.reactjs.org/>
6. Angular – <https://angular.io/>

Additional:

-

Breakdown of average student's workload

	Hours	ECTS
Total workload	100	4,00
Classes requiring direct contact with the teacher	60	2,50
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	40	1,50