

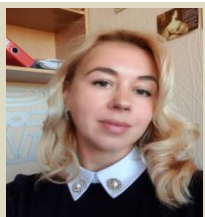
# FUNDAMENTALS OF WEB DEVELOPMENT

## COURSE SYLLABUS

<b>Code and name of specialty</b>	121 Software Engineering	<b>Institute</b>	Faculty of Computer Science and Software Engineering
<b>Program name</b>	"Software Engineering"	<b>Department</b>	Software Engineering and Management Information Technologies
<b>Type of program</b>	Educational and Professional	<b>Language of instruction</b>	Ukrainian, English

## LECTURER

**Name and surname, email** *YuliiaLitvinova, Uliya.Litvinova@kphi.edu.ua*



Ph.D. (01.05.02 Mathematical Simulation and Methods of Calculation). Associate Professor at the Department of Software Engineering and Management Information Technology. Work experience – since 2006. Author (co-author) of more than 40 research papers and textbooks. (h-index = 3 in Google Scholar - <https://scholar.google.com.ua/citations?user=8cVqocUAAAAJ&hl=uk>; ORCID ID is <https://orcid.org/0000-0001-6680-662X>)

Basic courses: "Fundamentals of Web development" (lectures and lab classes), Strategy of information systems "(lectures and lab classes), "Production organization and marketing" "(lectures and lab classes).

## GENERAL DESCRIPTION OF THE COURSE

<b>Summary</b>	The course "Fundamentals of Web Development" is a course in the cycle of professional compulsory training in the specialty 121 "Software Engineering". It is taught in the fourth semester in the amount of 90 hours (3 ECTS credits), in particular: lectures –32 hours, laboratory classes – 32 hours, independent work –26 hours. Course works are considered as individual tasks. The study of the discipline ends with the exam.						
<b>Course objectives</b>	Formation of students' theoretical and practical knowledge on the fundamentals of site design and web technologies; receiving practical skills in contemporary web programming.						
<b>Types of classes and control</b>	Lectures, workshops, consultations. The course ends with a final exam						
<b>Term</b>	4						

<b>Student workload (credits) / Type of course</b>	3 / Mandatory (elective)	<b>Lectures (hours)</b>	32	<b>Workshops (hours)</b>	32	<b>Self-study (hours)</b>	26
--	--------------------------	-------------------------	----	--------------------------	----	---------------------------	----

<b>Program competences</b>	<p>GC 05. Ability to learn and master modern knowledge.</p> <p>GC 06. Ability to search, process and analyze information from various sources.</p> <p>PC13. Ability to identify, classify and formulate software requirements.</p> <p>PC14. Ability to participate in software design, including modelling (formal description) of its structure, behavior and functioning processes.</p> <p>PC22. Ability to accumulate, process and systematize professional knowledge on the creation and maintenance of software and recognition of the importance of lifelong learning.</p> <p>PC25. Ability to reasonably select and master software development and maintenance tools.</p>
----------------------------	---

<b>Learning outcomes</b>	<b>Teaching and learning methods</b>	<b>Forms of assessment (continuous assessment CAS, final assessment FAS)</b>
--------------------------	--------------------------------------	--

PLO 1. Analyze, purposefully search for and select the necessary information and reference resources and knowledge to solve professional problems, taking into account modern advances in science and technology.

PLO 3. Know the basic processes, phases and iterations of the software life cycle.

PLO 6. Ability to select and use the appropriate task methodology of software development.

PLO 7. Know and apply in practice the fundamental concepts, paradigms and basic principles of operation of language, tools and computing software engineering.

PLO 8. Be able to develop a human-machine interface.

PLO12. Put effective approaches to software design into practice.

PLO14. Put into practice the tools of domain analysis, design, testing, visualization, measurement and documentation of software.

PLO 15. Being motivated to choose programming languages and development technologies to solve problems of software design and maintenance.

PLO23. Be able to document and present the results of software development.

Interactive lectures with presentations, discussions, laboratory classes, teamwork, case method, student feedback method, problem-based learning

Written individual assignments for laboratory works (CAS), assessment of knowledge in laboratory classes (CAS), express surveys (CAS), online tests (CAS), final/semester control in the form of a semester exam, according to the schedule of the educational process (FAS)

### ASSESSMENT AND GRADING

	Total score (points) for all types of learning activities	ECTS grading scale	The national grading scale	
Ranges of points corresponding to grades	90-100	A	excellent	Allocation of grade points
	82-89	B	good	
	74-81	C		
	64-73	D	satisfactory	
	60-63	E		
	35-59	FX		
	0-34	F	Unsatisfactory (with mandatory repetition of the course)	
			100% Final assessment as a result of Final exam (30%) and Continuous assessment (70%). 30% Final exam 70% Continuous assessment: Module №1 (7.5%) Module№2 (7.5%) Laboratory works (40%) Laboratory work №1 (8%) Laboratory work №2 (8%) Laboratory work №3 (8%) Laboratory work №4 (8%) Laboratory work №5 (8%)	

**Course policy** Students must attend all classes according to the study schedule and adhere to the norms of academic ethics. To study the course, students need to have their personal computer and (or) use computers of the computer center at the department. Students must work with compulsory and recommended reading, including Internet resources. Students must complete and submit all laboratory works during the semester in which the course is taught, before the examination session. The final assessment is not carried out without the personal presence of students.

## COURSE STRUCTURE AND CONTENT

<b>Topic 1</b>	Introduction to the fundamentals of web programming.			<b>Self-study</b>	WWW architecture: client/server architecture of the Internet. Overview of Web technologies. Web standards
<b>Topic 2</b>	Hypertext HTML language	<b>Laboratory work 1</b>	HTML. Fundamentals of hypertext markup. <i>Innovation Campus: Web FullStack-Sprint00</i>		Frames work
<b>Topic 3</b>	Using CSS cascading style tables.	<b>Laboratory work 2</b>	Cascading style tables. Practical use of CSS. <i>Innovation Campus: Web FullStack-Sprint00, Sprint01</i>		Operations and control structures. Functions and their parameters.
<b>Topic 4</b>	JavaScript language for client scripts.	<b>Laboratory work 3</b>	Dynamic HTML. Objects JavaScript. Practical usage the Web Forms. <i>Innovation Campus: Web FullStack-Sprint02</i>		Web development technologies for designing web-based information systems. Web-servers in information systems and their configuration.
<b>Topic 5</b>	The language of server scripts	<b>Laboratory work 4</b>	Server add-ons. PHP language. <i>Innovation Campus: Web FullStack-Sprint07</i>		Supervision of web-based information systems
<b>Topic 6</b>	The MySQL database management system. Principles of working with Internet databases	<b>Laboratory work 5</b>	Development of the web-interface to the database. A joint use of PHP and MySQL <i>Innovation Campus: Web FullStack-Sprint08</i>		

## RECOMMENDED READING

**Compulsory**

1. Terry Ann Felke-Morris. (2017). Web development And Desin Foundations with HTML5 (8 ed.).
2. Gaucher, J. D. (2019). HTML5. For professionals. Moscow: Peter.
3. John Duckett. (2019). Fundamentals of web programming using HTML, XHTML and CSS. Moscow: Exmo.
4. Jeremy, Keith. (2016). HTML5 for web designers. - Moscow: Mann, Ivanov & Ferber.
5. Josie, V. (2017). HTML for geography. How Google Earth works. Moscow: DMK Press.
6. Steve Suehring, Tim Converse, Joyce Park. (2010). PHP 6 and MySQL 6 Bible. (1st ed.). Wiley Publishing Inc.
7. Dronov, V. (2016). HTML 5, CSS 3 and Web 2.0. Development of modern Web-sites. Moscow: BHV-Peterburg.

**Recommended**

1. Sheldon, R. (2017). MySQL: a basic course. Dialectics.
  2. Dakett, J. (2018). HTML and CSS: Design and Build Websites.
  3. Valentine, C. (2011). XHTML. Williams Publishing House.
  4. Darnell, R. (2020). JavaScript. Peter.
  5. Zeldman, D. (2015). Web-design by standards. NT Press.
  6. Morrison, M. (2015). HTML and XML. Shvidkoiefektivno. Peter.
  7. Nielsen, J. (2020). Web-design. Symbol Plus.
- INFORMATION RESOURCES ON THE INTERNET
1. Retrieved from [www.microsoft.com](http://www.microsoft.com)-Microsoft website.
  2. Retrieved from [www.intuit.ru](http://www.intuit.ru)-Internet-Institute of Information Technology.
  3. Retrieved from [www.softtime.ru/bookphp/gl1\\_1.php](http://www.softtime.ru/bookphp/gl1_1.php) - PHP tutorial 4.
  4. Retrieved from [www.mysql.ru/docs/man/](http://www.mysql.ru/docs/man/)-reference manual for MySQL.
  5. Retrieved from [www.php.net](http://www.php.net)-PHP: Hypertext Preprocessor.
  6. Retrieved from [php.rus-phpnuke.com/](http://php.rus-phpnuke.com/)-PHP tutorial.
  7. Retrieved from [html.manual.ru/](http://html.manual.ru/)-HTML tutorial.
  8. Retrieved from [www.spravkaweb.ru/](http://www.spravkaweb.ru/)-Web languages guide.
  9. Retrieved from <http://htmlbook.ru/HTMLBook>-HTML handbook.
  10. Retrieved from <http://jquery.page2page.ru/>-jQuery manual

**ACADEMIC INTEGRITY**

Students are expected to adhere to the Code of Ethics of Academic Relations and Integrity of NTU “KhPI”.

The content of this syllabus is consistent with the course program.