



Syllabus Course Program

Development of complex applications by use of .NET

Specialty

122 – Computer science

Institute

Institute of Computer Modeling, Applied Physics and Mathematics

Educational program

Computer science. Modeling, design, and computer graphics

Department

Computer modeling of processes and systems

Level of education

Master's level

Course type

Professional training, Optional student disciplines of the profile training

Semester

2

Language of instruction

English

Lecturers and course developers



Marusenko Oleksii

oleksii.marusenko@khpi.edu.ua

Assistant of the Department of Computer Modeling of Processes and Systems
Author and co-author of more than 30 scientific and methodical publications.
Courses: "Development of complex applications by use of .NET", "Object-oriented programming C#".

General information, number of publications, main courses, etc.

Learn more about the teacher on the department's website

[More about the lecturer on the department's website](#)



Ivanchenko Kseniia Viktorivna

ksenii.ivanchenko@khpi.edu.ua

Candidate of technical sciences, associate professor, associate professor of the Department of Computer Modeling of Processes and Systems

Author and co-author of more than 30 scientific and methodical publications.
Courses: "Design of software systems", "Cross-platform programming", "Analysis and testing of software systems", "Programming of mobile devices".

[More about the lecturer on the department's website](#)

General information

Summary

The course contains the basics of programming using a software platform. NET. Creation of projects using the C# language and the .NET framework. Acquisition of programming skills using the main directions (multiple language support, cross-platform, use of class libraries, involvement of various technologies, productivity).

Course objectives and goals

Learning the knowledge, skills and abilities necessary to independently creating applications using .NET, as well as using the main directions and advantages. Acquisition of skills in the application of modern development approaches using special tools.

Format of classes

Lectures, laboratory classes, self-study. Final control in the form of a credit.

Competencies

PC01. Understanding of the theoretical foundations of computer science.

PC06. Ability to apply existing and develop new algorithms for solving problems in the field of computer science.

PC07. Ability to develop software in accordance with the formulated requirements, taking into account available resources and constraints.

PC08. Ability to develop and implement software development projects, including in unpredictable conditions, with unclear requirements and the need to apply new strategic approaches, use software tools to organize teamwork on the project.

PC11. Ability to initiate, plan and implement processes for the development of information and computer systems and software, including its design, analysis, testing, system integration, implementation and maintenance.

Learning outcomes

LO1. Have specialized conceptual knowledge, including modern scientific achievements in the field of computer science and is the basis for original thinking and research, critical thinking of problems in the field of computer science and on the border of knowledge fields.

LO6. Develop a conceptual model of an information or computer system.

LO17. Identify and eliminate problem situations during software operation, formulate tasks for its modification or reengineering.

LO18. Collect, formalize, systematize and analyze the needs and requirements for the information or computer system being developed, operated or maintained.

LO19. Analyze the current state and global trends in the development of computer science and information technology.

Student workload

The total volume of the course is 120 hours (4 ECTS credits): lectures - 32 hours, laboratory classes - 16 hours, self-study - 72 hours.

Course prerequisites

Basic knowledge of programming, understanding of the basics of object-oriented programming and design, mastery of programming technology in one or more programming languages

Features of the course, teaching and learning methods, and technologies

Using the free Microsoft Visual Studio software with the .Net Framework.

Program of the course

Topics of the lectures

Topic 1. Introduction.

Basics of the C# language and the .NET platform. The role of the platform. Managed and unmanaged code. JIT compilation.

Topic 2. Peculiarities of development and architecture of Web applications.

Development of ASP .NET and ASP.NET MVC (Model-View-Controller) web applications. Architecture of Web applications.

Topic 3. .NET Framework architecture.

Compilation and the MSIL language. Architecture of ASP.NET. ASP.NET State Management.

Topic 4. ASP.NET MVC Framework.

ASP.NET MVC Framework is a platform for Web applications. MVC scheme. The result of using MVC. Typical mistakes when using MVC.

Topic 5. Logic overload.

The controller is overloaded with the business logic of the Model. The model is overloaded with the logic of user representation.

Topic 6. Creating the first application.

Creating an MVC Web Application. An example of creating the first ASP.NET MVC application. The concept of Razor.

Topic 7. Fundamentals of routing.

Setting the route. The result of the application. Controller and Model (on the example of the store project). Answer (shop project). ASP.NET MVC strengths and weaknesses. ASP.NET Forms strengths and weaknesses.

Topic 8. Routing.

ASP.NET MVC Routing. Determination of routes. RegisterRoutes method. Default route definition, RouteConfig.cs file. Routes.MapRoute method parameters. Examples

Topic 9. Work with routes.

Creation of new routes. Mapping requests to files on disk. Procedure for determining new routes. Use of prefixes.

Topic 10. Parameters.

Receiving the passed parameters. Passing any number of parameters in the request. Creating restrictions for routes. Create your own restrictions. Ignoring requests.

Topic 11. Introduction to AJAX.

Setting up unobtrusive JavaScript/AJAX. AJAX helpers. AJAX forms. AjaxOptions object options.

Topic 12. Options of the AjaxOptions object.

LoadingElementId and LoadingElementDuration. Confirm. InsertionMode. Callbacks: OnBegin, OnSuccess, OnComplete, OnFailure. AJAX link. JSON format. AJAX requests using jQuery.

Topic 13. jQuery and AJAX.

Using JavaScript/jQuery. List of main scripts. Connection Using scripts on the master page. A quick overview of jQuery

Topic 14. jQuery function.

jQuery selectors. jQuery filters. Content filters. Form filters. jQuery Events. jQuery methods.

Topic 15. Web services: concepts and protocols.

Concept of web services. Web service protocols. SOAP introduction. Building blocks of SOAP. Skeleton of SOAP messages. Envelope element. SOAP header (Header). Characteristic MustUnderstand, Actor, encodingStyle. SOAP Body, SOAP Fault elements. Nested elements. Fault codes.

Topic 16. Introduction to WSDL.

WSDL description of web services. WSDL documents, basic structure and elements. WSDL PortType. Types of operations. Unilateral operation. Question-answer. WSDL binding. Binding to SOAP. WSDL and UDDI. Expanding web service technology stacks. Layers.

Topics of the workshops

Not included in the curriculum

Topics of the laboratory classes

Laboratory class 1.

C# language and .NET platform. Getting started in the software. Creating a new project and configuring configurations. The main elements of the working window of a new project.

Creating a primitive console program. Testing a new project. Running the first compiled application

Laboratory class 2.

Compile on the command line with the .NET CLI. Creating a .NET CLI project and working directory through command-line interaction. Using basic commands.

Editing the configuration file and application source code. Launching a test project. Search and launch the test application.

Laboratory class 3.

Program structure. Program execution. Instructions. Register dependence. Comments. Project files. Creating the first application on ASP.NET MVC 4. Creating a project. Conditions when creating models. Creating a data context and EntityFramework. Create a controller and view. Basics of routing. Data for default models. Stylization of the program. Compile, run, test.

Laboratory class 4.

Creating an Entity Framework data model for an ASP.NET MVC application. Creating an MVC web application. Setting the style of the site. Creating a data model. Setting up entities. SQL Server Express LocalDB. Setting up the seed method. Creating a controller and views. Project launch.

Laboratory class 5.

Implementation of basic CRUD functionality with Entity Framework in an ASP.NET MVC program. Creating the Details page. Creation of the Create page. Creating an Edit page. State of entities and methods Attach and SaveChanges Methods. Creating a Delete page. Checking connections to the database. Verification of the implementation of simple CRUD operations.

Laboratory class 6.

JavaScript and AJAX. Linking JavaScript/jQuery files. JavaScript/AJAX settings. AJAX helpers: Ajax.ActionLink, Ajax.RouteLink, Ajax.BeginForm, Ajax.BeginRouteForm, Ajax.GlobalizationScript, Ajax.JavaScriptStringEncode. Marking of forms.

Laboratory class 7.

Working with jQuery UI and visual elements. jQuery UI and Autocomplete. Connecting jQuery UI libraries (Combined Library). Connecting basic styles. An example of the Autocomplete widget. Working with the HomeController. Plugin testing

Laboratory class 8.

Using web services in ASP.NET's AJAX technology. Options for use. Basic information. Examples. Additional resources. References to classes. New opportunities. Client-server interaction for AJAX clients. Client architecture of AJAX technology. AJAX server architecture. Calling methods of web services in AJAX. Making HTTP requests from an AJAX client script. Calling WCF service operations in AJAX.

Self-study

1. Independent work with theoretical material.
2. Implementation of the examples given in the theoretical part.
3. Report on the creation of an independent application for the selected preliminary topic and set of functions with further demonstration of functionality.

Course materials and recommended reading

1. Free. Cross-platform. Open source. A developer platform for building all your apps. - Access mode: <http://msdn.microsoft.com/net>.
2. Guide to ASP.NET MVC 5. – METANIT.COM programming site. – Access mode: <http://metanit.com/sharp/mvc5/>.
3. Grimes F. Microsoft .NET for Programmers./ F. Grimes - Manning PublKes, 2002, - 386 pp.
4. Richter J. Applied Microsoft .NET Framework Programming./ J. Richter - Microsoft Press, 2002, - 556 pp.
5. Visual C#. NET Step by Step, Microsoft Press, 2003. - ISBN: 0-7356-1909-3.
6. Troelsen E. The C# 9 programming language and the .NET 5 platform: basic principles and programming practices / E. Troelsen, F. Jepiks - 10th edition, 2021 -pp. 770 - ISBN 978-617-7987-81-8
7. Freeman A. ASP .NET MVC 5 with examples in C# 5.0 for professionals. / A. Freeman - C. 736 pages - ISBN 978-5-8459-1911-3
8. Wentz K. Programming in ASP.NET AJAX / K. Wentz -2017, p. 502.
9. Freeman A. ASP. NET MVC 3 Framework with examples in C# for professionals / A. Freeman, S. Sanderson - Williams, 2012, - C. 672 - ISBN 978-5-8459-1758-4.

Assessment and grading

Criteria for assessment of student performance, and the final score structure

The processing of questions assigned for self-study is evaluated by the lecturer during the credit at the end of the academic semester.

Points are awarded according to the following ratio:

- Report: 40% of the semester grade
- laboratory works: 60% of the semester grade).

Grading scale

Total points	National	ECTS
90-100	Excellent	A
82-89	Good	B
75-81	Good	C
64-74	Satisfactory	D
60-63	Satisfactory	E
35-59	Unsatisfactory (requires additional learning)	FX
1-34	Unsatisfactory (requires repetition of the course)	F

Norms of academic integrity and course policy

The student must adhere to the Code of Ethics of Academic Relations and Integrity of NTU "KhPI": to demonstrate discipline, good manners, kindness, honesty, and responsibility. Conflict situations should be openly discussed in academic groups with a lecturer, and if it is impossible to resolve the conflict, they should be brought to the attention of the Institute's management.

Regulatory and legal documents related to the implementation of the principles of academic integrity at NTU "KhPI" are available on the website: <http://blogs.kpi.kharkov.ua/v2/nv/akademichna-dobrochesnist/>

Approval

Approved by

Date, signature

Head of the department
Dmytro BRESLAVSKY

Date, signature

Guarantor of the educational program
Oleksii VODKA