



UDC 004.42:004.77

DEVELOPMENT OF A WEB-BASED SYSTEM FOR CONTENT EXCHANGE

РОЗРОБКА ВЕБ-СИСТЕМИ ДЛЯ ОБМІНУ КОНТЕНТОМ

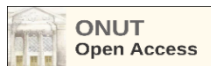
Yevhenii Kushnir
Євгеній Кушнір

State University of Trade and Economics, Ukraine, Kyiv

ORCID: <https://orcid.org/0009-0003-5862-9223>E-mail: y.kushnir_fit_5m_23_m_d@knote.edu.ua

Copyright © 2024 by author and the journal “Automation of technological and business – processes”.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0>DOI: <https://doi.org/10.15673/atbp.v16i2.2852>

Abstract. The research paper presents a comprehensive study and development of a web-based content sharing system, the main purpose of which is to stimulate and distribute photo and media content, textual information about video game characters in the Ukrainian segment of the Internet through a convenient and efficient platform for users. This study formulated the requirements for the content sharing system and the stages of its development. Based on the requirements for the implementation of the system, a web-based content exchange system was developed, consisting of a client application (frontend) and a web server (backend) using third-party SaaS solutions to store media content and user information for the content exchange system. The system is designed to provide users with a convenient and efficient tool for sharing information about video games. The development process consists of the following stages: researching the context of formulating functional and non-functional requirements, selecting technologies to implement the system in accordance with the requirements for the frontend and backend. Also, unit and end-to-end tests are written for the developed applications to ensure that the system has the expected behavior. Completion of all stages and implementation of front-end and back-end applications in accordance with the requirements is the result of this work, namely a web-based content exchange system that meets all the requirements and provides an effective exchange of information about video games in the Ukrainian Internet segment.

Анотація. У дослідницькій роботі представлено комплексне дослідження та розробку веб-системи обміну контентом, основною метою якої є стимулювання та розповсюдження фото- та медіа-контенту, текстової інформації про персонажів відеоігор в українському сегменті Інтернету через зручний та ефективна платформа для користувачів. У цьому дослідженні сформульовано вимоги до системи обміну контентом та етапи її розробки. Виходячи з вимог до впровадження системи, була розроблена система обміну веб-контентом, яка складається з клієнтської програми (frontend) і веб-сервера (backend) з використанням сторонніх рішень SaaS для зберігання медіа-контенту та інформації користувача для система обміну контентом. Система розроблена, щоб надати користувачам зручний і ефективний інструмент для обміну інформацією про відеоігри. Процес розробки складається з наступних етапів: дослідження контексту формулювання функціональних і нефункціональних вимог, вибір технологій реалізації системи відповідно до вимог до фронтенду та бекенда. Крім того, модульні та наскрізні тести написані для розроблених програм, щоб переконатися, що система має очікувану поведінку. Результатом цієї роботи є завершення всіх етапів і впровадження фронтенд і бекенд додатків відповідно до вимог, а саме веб-система обміну контентом, яка відповідає всім вимогам і забезпечує ефективний обмін інформацією про відеоігри. в українському сегменті Інтернету.

Keywords: Web-oriented systems, content exchange system, frontend, backend, requirements for the content exchange system, unit and end-to-end testing, third-party SaaS solution.

Ключові слова: Веб-орієнтовані системи, система обміну контентом, фронтенд, бекенд, вимоги до системи обміну контентом, модульне та наскрізне тестування, стороннє рішення SaaS.

I. INTRODUCTION

A new tweet, a message in a messenger, participation in an online conference, watching a popular video, leaving a comment on a post - all of these are the result of a person's continuous integration into content-sharing systems. Such



systems differ in their implementation and specifics of use, but they all have a common feature - they provide the ability to distribute and receive any information regardless of the user's location and time.

Content-sharing systems are directly created for Android, Windows, and Linux platforms, but the most widespread, thanks to the Internet, are Web-oriented systems. Since they are not tied to the platform on which they run, the number of users who will use such a system is much higher compared to other implementations of such a system.

The research topic is determined by its importance in exploring and promoting a new niche for the Ukrainian segment. The web application is aimed at a specific type of content, namely textual information, video and photo galleries. Creating a content-sharing system is a creative and detailed process that requires not only knowledge of a technical discipline but also confident knowledge of the theoretical side and its application in practice.

II. LITERATURE ANALYSIS

2.1. Approaches to creating a web application

To create a web application, a developer in the "JavaScript world" has to choose from a large number of options, not only types of web applications but also the technology used to create the application. This is because JavaScript is one of the most widespread programming languages, as well as the ability of anyone to create and distribute their solution to a problem or a new solution as a Node Package Module.

The definition of "web application" can be perceived differently depending on the environment where web technologies are used. For example, a web application can be created as a full-fledged Android, Desktop, web server, or console application, but the main use case is to run in a browser.

The types of web applications include:

- Single Page Application (SPA): this is a type of application that runs only on the client side. The application is interactive for the user and navigation throughout the application takes place without reloading the page;
- Multi Page Application (MPA): traditional multi-page web applications. When switching between pages, the user loads a new page;
- Progressive Web Application (PWA): similar features, functions, and quality of user experience to traditional computer and mobile applications based on SPA solutions. However, it is difficult to distinguish such applications as non-PWA or PWA. PWA applications are characterised by the following aspects, they must have a proxy layer (Service Worker) and a Web App manifest.

Popular web technologies used to create a web application are React, Vue, and Angular. These technologies can also use the SPA and PWA approaches, combining them to improve the user experience.

In addition to these technologies, some use one of the already mentioned technologies for page rendering, use the Server Side Rendering (SSR) approach for SEO page optimisation, speed up page loading speed, and provide an opportunity to build an application using all three types of web applications.

In cases where a web application does not require the use of any of the above technologies, but rather pure JavaScript solutions, use builders such as Vite, Webpack, Rollup, Parcel; Gulp. Most of these technologies use a bundler to improve the developer experience.

It is also worth noting that there are systems for creating web applications, namely content management systems, or CMS (Content Management System), which is a website content management system, or they are also called CMS engines [1]. Such systems offer the use of ready-made templates for a website, provide ready-made functional solutions for managing website content and design, and various plugins that implement certain functionality for a website. Examples of CMS are WordPress, Joomla, OpenCart, Shopify, and Fix.

2.2. Business requirements for a content exchange system

Defining precise requirements ensures that all parties involved in the creation of the system have a clear understanding of what they are developing.

Business requirements define the strategic path of the project. They include high-level project requirements that describe the overall business goal from the company's perspective. The goal may be to increase profits or capture a larger market share [2]. The business requirements of the content sharing system are to provide the user with the opportunity to get acquainted with the character's story and multimedia content (photos, videos) from games and other entertainment services, interact with the user to make him or her interested in the content offered and promote a new niche in the Ukrainian segment by creating a united community.

Solution requirements describe the specific characteristics that a product must have to meet the needs of stakeholders and the business itself. They fall into two broad groups [3]:

- Functional requirements describe the capabilities that a solution should have in terms of behaviour and information that the solution will manage [4];
- Non-functional requirements are not directly related to the behaviour of the solution's functionality, but rather describe the conditions under which the solution must remain effective or the qualities that the solution must have [4].

Another important stage is the creation of a project solution. A project solution is a reasoned recommendation for achieving a specific desired result, determining its feasibility for implementation in the form of a project. It is an integral part of the project and correlates with it as a part of the whole.

2.3. Technology stack for developing a content exchange system

In web development, choosing the technologies that will be used to develop the software implementation is always an important stage because in the world of JavaScript, there are a lot of solutions that solve specific requirements in an extremely efficient way.



A technology stack is a set of technologies that are put together to create any application. Also known as a technology infrastructure stack or solution stack, a technology stack has become essential for building easy-to-maintain, scalable web applications [5]. The technology stack of a content exchange system consists of two important elements: the front-end (client side) and the back-end (server side).

The success of the business project will also depend on the right choice of the technology stack. Therefore, when choosing a technology stack, you should focus on the following key points:

- Workflow support: Implementing new software solutions can be time-consuming. The team needs to adapt, and it takes time to increase efficiency. However, choosing the right set of technologies from the outset can help avoid unnecessary changes in the future [6];
- Budget savings: rebuilding a technology stack is costly. This may include reworking applications, reviewing and changing processes, and even hiring new employees with specialised skills [6];
- Supporting innovation and scaling: budget and team staffing constraints can slow down product development, and while the technology stack should not be allowed to be the main reason for pursuing new opportunities and ideas, it should always be ensured that the technology stack encourages innovation rather than restricts it [6].

Following the defined requirements (see section 4.1) and project decision stages (see section 4.2), the successful implementation of a web application requires

- Create a website layout;
- Implement a web server using third-party services and create API documentation;
- Implement a web application according to the layout.

Therefore, before proceeding to the implementation of a web application, you need to determine which technologies should be used at each of the above stages.

To create a layout, you can use Figma, Axure RP, Draftium, InVision, or Adobe XD. In the implementation of the target system, the Figma tool will be used, due to a small entry curve for starting to use, a sufficient amount of functionality, and no restrictions on creating and editing projects.

To develop a web server, you can use Spring MVC, Express, Laravel, and Django. In the implementation of the target system, the Express framework will be used. The advantages of this technology are a single programming language for a web application and a web server, ease of use, and a large number of extensions and packages for solving various tasks.

When choosing a technology for the web server, you need to select a service that will provide SDK solutions for interacting with the database and cloud media storage and a service for sending emails.

For the web server, we will use the Firebase SDK solution, as the above-mentioned services provide a large amount of functionality, but have a different entry curve for starting to use them. Compared to AWS and Microsoft platforms, Firebase does not require additional knowledge of the platform itself and its infrastructure to be used.

Mailchimp, Sendinblue, and GetResponse can be used for sending emails. In the implementation of the target system, Sendinblue will be used, due to the ease of use of the service and the largest number of messages provided in the free tariff.

For API documentation, Swagger will be used, as Swagger has no disadvantages among all other technologies, and also provides self-generated documentation for API documentation without the need to change the development environment.

To create a web application, the Webpack builder will be used, due to the possibility of greater control over the build configuration process and ease of extension if necessary to add a web framework or other technology compared to other builders such as Vite, Rollup, and Parcel.

Following the requirement for the webserver to use the SSR approach and the web application and web server technology stack, Handlebars will be used to create dynamic and static HTML pages. This technology will reduce the amount of repetitive code by splitting the template into separate parts and reusing them, and the technology will provide a single solution for the client and server sides. SCSS will be used to style the pages, which is a preprocessor scripting language that is a superset of CSS.

III. OBJECT, SUBJECT, AND METHODS OF RESEARCH

The object of research is the process of developing Web-oriented systems.

The subject of the study is Web-oriented content exchange systems as a means of obtaining and disseminating information and technologies for their development.

Research methods. In this paper, the theoretical basis of the research is:

- Methods of analysing literature and scientific sources: a review of publications related to the development of a content exchange system;
- Methods of requirements management: determination of user requirements, namely functional and non-functional requirements, formation of a design solution;
- Algorithmic programming methods for creating web application and web server functionality;
- Analysis methods for selecting appropriate technologies, platforms for developing a content exchange system.

IV. RESULTS

4.1. Requirements for a content exchange system

The functional requirements for a content exchange system should include the following capabilities:

- Creating a user account by registering;
- User authorisation;



- View multimedia content such as photo gallery, video and text information on the "character" page for an authorised user;
- Delete multimedia content (photos) from the "favourite photos" category;
- Add multimedia content (photos) to the "favourite photos" category;
- Create a personal list of "favourite characters";
- Select an account profile photo from the list;
- Edit personal account information;
- Filter photo content by criteria;
- Send an email to the developer.

Non-functional requirements for a content exchange system:

- Requirements for use:
 - The system should be web-based;
 - The design of the web application should be based on the "Desktop first" approach.
- Performance requirements: the system must support several thousand simultaneous requests to the web server and a third-party service for storing media content and user information.
- Requirements for implementation:
 - Programming languages for the implementation of the web application and web server should be JavaScript and TypeScript;
 - The architectural pattern for designing and developing a web server should be MVC (Model View Controller);
 - The architectural pattern for designing and developing a web application should be MVVM (Model View ViewModel);
 - The project should be built using a monolithic architecture;
 - The web server should use the SSR (Server Side Rendering) approach for dynamic pages;
 - A bundler should be used for the web application, namely static pages;
 - Use a third-party service to store media content, user information, and other information required by the system. The service should provide SDK (Software Development Kit) solutions for working with the database and for working with cloud storage for media content;
 - It should be possible to edit, add and change media content in the system only through the use of a third-party control panel to save all media resources;
 - The minimum version of browsers in which the web application should work is: Chrome 49, Firefox 18, Safari 10, Edge 12.
- Reliability requirements:
 - The web server must be able to withstand a simultaneous load of at least 2000 users;
 - The web application should provide the user with only predefined functionality.
- Interface requirements:
 - The web application must interact with the web server via the REST API (Representational State Transfer Application Programming Interface);
 - The web server should interact with a third-party service for storing media resources and user information via SDK.
- Security requirements:
 - Routes and APIs that require user authentication should be protected on the web server side;
 - The web server must use CSRF (Cross-Site Request Forgery) protection against cross-site request forgery; Authorised users can only change data that is directly related to them.

4.2. Design solution for a content exchange system

The project solution for a content exchange system will be a set of incremental stages that determine the amount of work that needs to be done to successfully implement the project.

Stages of developing a content exchange system:

- Research stage:
 - Market and competitor research: comparing the capabilities of competitors and the target system (see Table 1).
 - Analysis of the target audience: following the "5W" method:
 - "What": the opportunity for users to get acquainted with multimedia content and textual information about the world of games and other entertainment services and interactive interaction with the user;
 - "Who": people of any age who like to spend time playing games;
 - "Why": the ability to quickly get acquainted with certain content;
 - "Where": search engines;
 - "When": any time, mostly in the evening after 19:00.
- Stage "Development of a full-fledged website design":
 - Designing the interface and architecture of the website;
 - Development of sections and pages for the website layout.
- Stage "Website layout according to the design":
 - Layout of all pages according to the design;
 - Work with animation of elements on the pages.
- Stage "Implementation of the server part":



- Creating routing for the website;
- Creating public and private REST APIs;
- Creation of API documentation;
- Using third-party services to obtain user information and data about multimedia content.
- Stage "Testing the web application and web server":
- Writing Unit tests for web server services;
- Writing End-to-end tests for a web application.
- Support stage:
- Publishing the source code in a repository of one of the web services for version control system repositories (Git).

Table 1. Competitors' characteristics research

Features / Competitors	Fandom Website	«Вісник Паймон» Website	Honey Impact Website	Target system
Content diversity	+	-	-	+
Interactive user experience	+	+	+	+
Photo gallery	-	+	-	+
Create a user profile	+	-	+	+
Availability of the "favourite photos" category	-	-	-	+
Focus on the Ukrainian-speaking segment	-	+	-	+

4.3. Software implementation of the content exchange system

Following the steps for successful implementation of a web application (see Section 2.3), the first stage of implementing a content exchange system is to create a mockup in Figma. Following the requirements and the technology stack, a web application mockup was created, Fig. 1 shows the layout of the main page of the web application.

The following pages have been developed in the course of the site's layout:

- Home page - contains brief information about the service. The page contains minimal text of up to 100 words in black. All navigation elements and images on the page are intuitively simple for the user;
- The page "Taiwan through photos" implements the requirement "filtering photo content by criteria". On the right side of the page, there is a list of photos with thematic content, and on the left side, there is a functionality for direct filtering;
- Authorisation page - contains a form with fields for entering email and password, as well as a link to the registration page;
- Registration page - contains a form with fields for entering your email, password, name, selection of a game element (water, fire, earth, air, etc.) and a game city from the proposed list;
- User profile page - this page contains a list of photos that belong to the category of the user's "favourite characters" and the ability to edit the list by deleting certain photos. The page also provides an opportunity to edit the user's personal information and update the list of "favourite characters";
- Sending a message page - this page implements the requirement to send an email to the developer, it contains a form with a field for mail and a test message;
- Page with a list of characters - contains a list of all characters and a link to the detailed page with the character;
- Character page - contains detailed information about the character in the form of text information, video and photo content, and also has the functionality to add a photo to the "favourite photos" category;
- About page - contains a short text from the developer of the content sharing system, no more than 50 words.

The link to the entire layout for the web application is attached:

<https://www.figma.com/file/8gxl94XyHiuedslxtSG9y/Keqing-site?node-id=1305%3A15&mode=dev>.

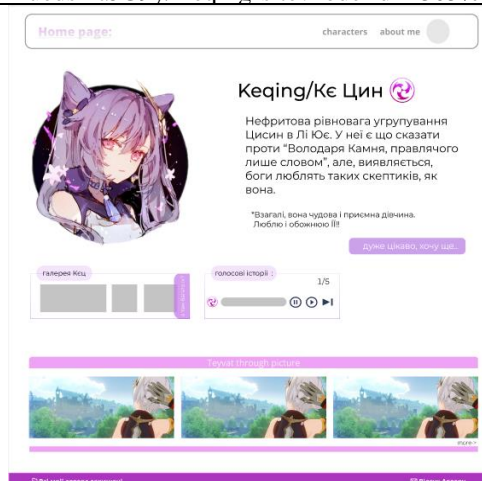


Fig. 1. Home page layout



The second stage involves the implementation of the web server itself using third-party services and the creation of API documentation. The requirements for the web server also include the use of the MVC architectural pattern and CSRF protection.

Fig. 2 shows the architecture of the web server following the MVC template and Fig. 3 shows the initialisation of CSRF protection in the web server.

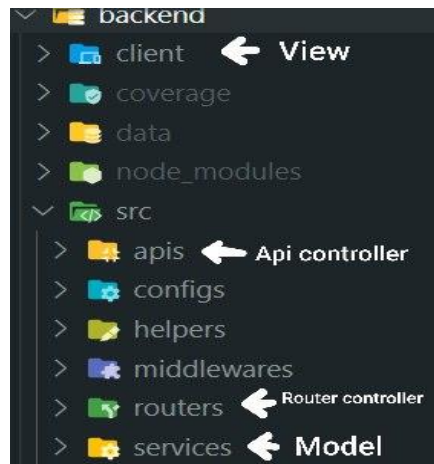


Fig. 2 Web server architecture

```
const csrf = require('csrf')
...
const csrfSecure = csrf({ cookie: true })
const csrfSecureAPI = csrf({
  cookie: true,
  ignoreMethods: ['HEAD', 'OPTIONS'],
})
...
app.use(csrfSecure)
...
app.all('*', (req, res, next) => {
  res.cookie('XSRF-TOKEN', req.csrfToken())
  next()
})
app.use(site_router)
if (process.env.NODE_ENV === 'test') {
  app.use('/api/v1', api_v1)
} else {
  app.use('/api/v1', csrfSecureAPI, api_v1)
}
```

Fig. 3. CSRF protection in a web server

Following the defined technology stack for implementation:

- The web server is based on the Express framework;
- The Firebase SDK was used to work with the database and cloud storage for media content;
- Sendinblue is used for sending emails;
- Handlebars is used for SSR approach;
- API documentation is based on the NPM package "swagger-ui-express". The web server provides access to the API documentation page via the following route "[URL]/api-docs".

The third stage is the implementation of the web application. Following the defined technology stack at this stage:

- We used the Webpack bundler with its configuration;
- Web pages are built according to the layout.

Another requirement for creating a web application is to use the MVVM architectural pattern. Fig. 4 shows the architecture of a web application using the MVVM template. Accordingly, after creating the configuration file for the bundler, the pages for the web application are built, Fig. 5 shows the view of the main page.



<http://www.atbp.ontu.edu.ua/>

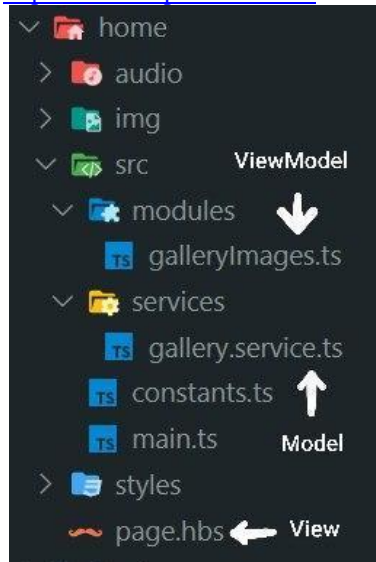


Fig. 4. Web application architecture

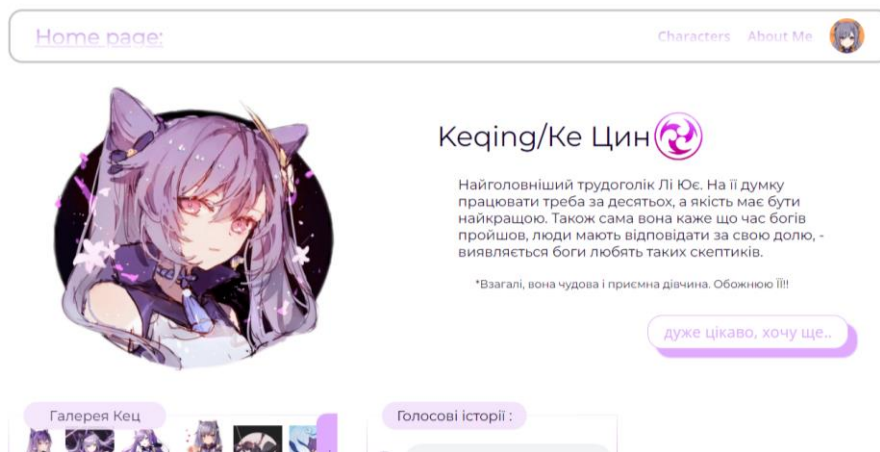


Fig. 5. Home page of the web application

4.4. Testing the content exchange system

Following the testing phase in section 4.2, to test the content exchange system, you need to write Unit tests for the services (business logic) of the web server and write End-to-end tests for the web application.

Jest is used to write unit tests. Fig. 6 shows the details of implementing a unit test of a service for working with a database. To start the test execution process locally, run the following command in the terminal of the project root folder "npm run be:test", Fig. 7 shows the result of running all unit tests.

```

it('#get_user', async () => {
  const res = await user_data.get_user(mockUserData.uid)

  expect(res).toEqual(mockUserData)
  expect(mockStoreCollection).toBeCalledWith('users')
})

it('#update_user', async () => {
  const newData = { newData: 'test' }

  await user_data.update_user(newData, mockUserData.uid)

  expect(mockStoreSet).toBeCalledWith({ ...mockUserData, ...newData })
})

it('#update_user_stand', async () => {
  const newData = { stand: 'test' }

  await user_data.update_user_stand(newData, mockUserData.uid)

  expect(mockStoreSet).toBeCalledWith({ ...mockUserData, stand: newData })
})

it('#update user favorite new item', async () => {

```

Fig. 6. Unit test of the service for working with the database



```

> keqing_site_backend@1.0.0 test
> jest

PASS src/services/firebase/auth.service.test.js (7.233 s)
PASS src/services/firebase/storage.service.test.js (7.243 s)
PASS src/services/firebase/store.service.test.js (7.39 s)
PASS src/services/local_data.service.test.js
PASS src/services/mail.service.test.js

Test Suites: 5 passed, 5 total
Tests: 15 passed, 15 total
Snapshots: 0 total
Time: 14.167 s
Ran all test suites.
    
```

Fig. 7. The result of the Unit tests

Cypress was used to write and execute E2E tests. Fig. 8 shows the details of implementing E2E tests for the character page. To start the process of running tests locally, run the following command in the terminal of the project root folder "npm run fe:test", Fig. 9 shows the result of the E2E test for the character page.

```

describe('Test character page', () => {
  beforeEach(() => {
    cy.visit('/login?continuePath=/characters/ayaka')
    cy.fixture('user-credentials.secret').then((user) => {
      cy.get('#email').type(user.email, { force: true })
      cy.get('#password').type(user.password, { force: true })
      cy.get('form button[type="submit"]').click()
    })
  })

  beforeEach(() => {
    cy.get('.k-header-controls-item.k-images-btn_open').as('openImages')
    cy.get('.k-images').as('listImages')
    cy.get('.k-header-controls-item.k-video-btn_open').as('openVideo')
    cy.get('.k-video').as('video')
  })

  it('#open character images should work', () => {
    cy.get('@openImages').click()
    cy.get('@listImages').should('be.visible')
  })

  it('#open character video should work', () => {
    cy.get('@openVideo').click()
    cy.get('@video').should('be.visible')
  })
})
    
```

Fig. 8. E2E test for a character page

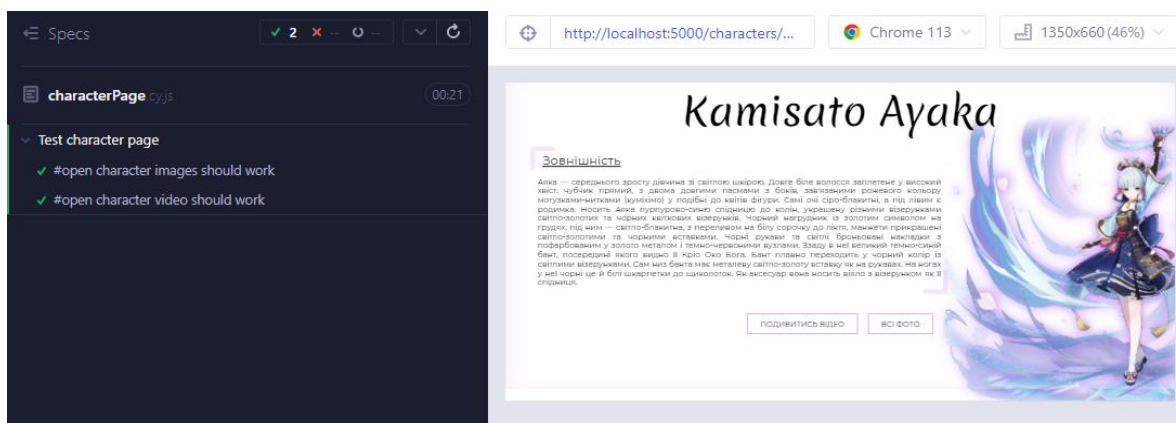


Fig. 9. The result of the test for the character page

V. CONCLUSIONS

The paper presents the results of a theoretical study and practical implementation, which consist of the development of a WEB-oriented content exchange system. The results of the theoretical study include the formation of functional and non-functional requirements and a design solution for the system. The results of the theoretical study became the basis for the practical implementation of a WEB-oriented content exchange system. As a result of the practical implementation of the content-sharing system, the following conclusions were obtained:

- The layout for the web application was implemented following the functional requirements, which made it possible to identify inaccuracies in the requirements, obtain the main features of the web application in advance and adjust the development process.
- The software implementation of the client part of the content exchange system was carried out following the created layout, defined requirements and the selected technology stack in the form of a multi-page web application.



<http://www.atbp.ontu.edu.ua/>

- Software implementation of the web server for the content exchange system following the requirements using third-party services, the selected technology stack and ensuring the protection of the web server from cross-site request forgery.
- We tested the content exchange system by writing Unit tests for the web server and End-to-end tests for the web application. Testing ensures that the system works correctly and directly helps to detect errors if they occur when a certain time of the system changes.

• To expand the possibility of project support and due to the large code base, the source code of the content-sharing system is hosted on the GitHub repository service, the link is attached: <https://github.com/zenia369/keqing-site>.

A web-based system for content sharing is available on the Internet and can be accessed at the following link: <https://keqing-site.onrender.com/>.

VI. REFERENCES

- [1.] 07Hostiq. What is a website CMS? URL: <https://hostiq.ua/wiki/ukr/cms/> (дата звернення: 07.04.2024)
- [2.] Altexsoft (2021, August 27). How to Write a Business Requirements Document: Guidelines, Templates, and Useful Tips. URL: <https://www.altexsoft.com/blog/business-requirements-document/> (дата звернення: 07.04.2024)
- [3.] Altexsoft (2023, November 30). Functional and Nonfunctional Requirements: Specification and Types. URL: <https://www.altexsoft.com/blog/functional-and-non-functional-requirements-specification-and-types/> (дата звернення: 07.04.2024)
- [4.] Brennan K. (2015). A Guide to the Business Analysis Body of Knowledge (BABOK Guide) 3rd edition. International Institute of Business Analysis. 512p.
- [5.] MongoDB. What is a Tech Stack and How Do They Work? URL: <https://www.mongodb.com/basics/technology-stack> (дата звернення: 07.04.2024)
- [6.] Azbyka. Technical stack. What is it? URL: <https://azbyka.com.ua/uk/tehnicheskij-stek1/> (дата звернення: 07.04.2024)

Отримана в редакції 10.05.2024. Прийнята до друку 31.05.2024. Received 10 May 2024. Approved 31 May 2024. Available in Internet 23 July 2024

UDK: 004.9

DEVELOPMENT OF A COMPUTER-BASED EDUCATIONAL PROGRAM ON THE SUBJECT "FUNDAMENTALS OF COMPUTER MODELING"

РОЗРОБКА КОМП'ЮТЕРНОЇ НАВЧАЛЬНОЇ ПРОГРАМИ З ДИСЦИПЛІНИ "ОСНОВИ КОМП'ЮТЕРНОГО МОДЕЛЮВАННЯ"

Kim Yekaterina¹, Alimbekova Ayaulym², Gavrilova Anastassiya³, Kan Alexandr⁴
Кім Єкатеріна¹, Алімбекова Аяулим², Гаврилова Анастасія³, Кан Олександр⁴

^{1,2,3,4}Turan University, Almaty city, Republic of Kazakhstan

ORCID: <https://orcid.org/0000-0001-7441-524X>¹, <https://orcid.org/0009-0001-6092-8762>²,

<https://orcid.org/0009-0000-5286-604X>³, <https://orcid.org/0009-0001-6092-8762>⁴

E-mail: e.kim@turan-edu.kz¹, 22230750@turan-edu.kz², 21220856@turan-edu.kz³, 21222052@turan-edu.kz⁴

Copyright © 2024 by author and the journal "Automation of technological and business – processes".

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0>



DOI: <https://doi.org/10.15673/atbp.v16i2.2851>

Abstract. In the conditions of technological progress and rapid development of information technologies, it is very challenging to prepare specialists in the field of information and communication technologies who would fully possess the knowledge and skills to work with the diverse range of instrumental and software tools and equipment used in the field. This is due to the continuous improvement and updating of programming languages, frameworks, software environments, shells, and hardware. One of the ways to address this issue is through computer-based educational programs, allowing the individualization of the learning process while preserving its quality. Currently, there is a large



number of computer-based educational programs used in the study of various disciplines. This paper presents the design of a computer-based educational program on the subject "Fundamentals of Computer Modeling," which enables the modeling of random patterns of various natures, ranging from random events to random streams. The educational program consists of three modules, which include a theoretical part on methods of modeling random patterns, testing after each topic, and a calculator for all modeling methods. In each modeling method, you can input any initial data, such as the number of trials, probabilities of events, functions, and parameters of distribution laws. The results of modeling random patterns can be saved as a file. The program interface is available in three languages: Russian, English, and Kazakh. The currently developing educational program aims to improve the learning process in the "Fundamentals of Computer Modeling" discipline and allows learners to study it independently without any interventions from a professor.

Анотація. В умовах технічного прогресу та стрімкого розвитку інформаційних технологій дуже складно підготувати спеціалістів у галузі інформаційно-комунікаційних технологій, які б у повній мірі володіли знаннями та навичками роботи з різноманітним спектром використовуваних інструментальних та програмних засобів та обладнання. в польових умовах. Це пов'язано з безперервним удосконаленням і оновленням мов програмування, фреймворків, програмних середовищ, оболонок і апаратного забезпечення. Одним із шляхів вирішення цієї проблеми є комп'ютерні навчальні програми, що дозволяють індивідуалізувати процес навчання, зберігаючи його якість. В даний час існує велика кількість комп'ютерних навчальних програм, які використовуються при вивченні різних дисциплін. У цій роботі представлено дизайн комп'ютерної навчальної програми з дисципліни «Основи комп'ютерного моделювання», яка дозволяє моделювати випадкові закономірності різної природи, починаючи від випадкових подій і закінчуючи випадковими потоками. Навчальна програма складається з трьох модулів, які включають теоретичну частину з методів моделювання випадкових закономірностей, тестування після кожної теми та калькулятор для всіх методів моделювання. У кожному методі моделювання можна вводити будь-які вихідні дані, такі як кількість випробувань, ймовірності подій, функції та параметри законів розподілу. Результати моделювання випадкових закономірностей можна зберегти у вигляді файлу. Інтерфейс програми доступний трьома мовами: російською, англійською та казахською. Навчальна програма, що розробляється, спрямована на вдосконалення навчального процесу з дисципліни «Основи комп'ютерного моделювання» та дозволяє слухачам вивчати її самостійно без втручання викладача.

Keywords: education informatization, educational process, computer-based educational program, instructional programs, simulation modeling, random patterns, continuous random variables, discrete random variables.

Ключові слова: інформатизація освіти, навчальний процес, комп'ютерна навчальна програма, навчальні програми, імітаційне моделювання, випадкові закономірності, неперервні випадкові величини, дискретні випадкові величини.

I. INTRODUCTION

In the era of information technology development, there is a wide variety of programming languages, frameworks, instrumental and software tools, shells, and equipment that students, enrolled in the educational programs "6B06101 - Information Systems" and "6B06103 - Intelligent Robotics," need to be familiar with. Typically, specialists in these fields must be able to use various types of instrumental and software environments and equipment.

The limited number of hours allocated to different disciplines and the constant evolution and updating of information technologies and equipment do not allow for structuring the educational process to cover all existing programming languages, instrumental and software tools, shells, and equipment.

For the quality and effective performance of tasks assigned to specialists in the field of information technology and robotics, they need to constantly improve their skills and explore new digital tools and equipment. This process should continue even after they have completed their education in this field.

One solution to this problem is computer-based educational programs aimed at self-learning, which, in turn, allow individuals to undergo training at a convenient (free) time for them.

II. LITERATURE ANALYSIS

More and more educators are utilizing various informational and interactive teaching methods and technologies [1-11]. The use of informational and interactive methods and technologies enables the activation of the cognitive activities of learners, optimizes the task execution process, and improves communication between the teacher and learners.

Currently, there are a multitude of digital tools that can be employed in teaching various disciplines. Among such tools are no-code tools, online compilers, integrated programming environments for software development, platforms for creating tests and quizzes, virtual trainers and simulators, educational platforms, etc.

Many researchers are involved in the development and use of computer-based educational programs for studying various disciplines. For instance, in [6], the authors develop educational and control programs, an electronic textbook "Forensic Studies of Narcotics and Psychotropic Substances" for students in the field of "Forensic Expertise," which includes two lecture presentations, two educational programs, six control tests, two instructional videos, and other educational materials covering all the topics studied.

In [7], the authors use an instructional (control and instructional) program in the study of the discipline "Digital Devices and Microprocessors," which is developed in the "Delphi" language.

When studying electronic devices and designing the structure of radio-electronic equipment, it is convenient to use various electronic laboratory practicals, as demonstrated in [8].



The authors of [9] developed five virtual laboratory works for studying the discipline "Heat Supply of Forest Industry Enterprises," which students can perform remotely. Thanks to the developed laboratory works, students reduce the time of their completion by accelerating simulated processes and can visually study and analyze processes under unlimited conditions. Virtual laboratories can also be used as an alternative to using outdated equipment during its repair or before purchasing new equipment.

In [10], the authors propose a computer-based educational program for studying the topic "Graphical Method for Solving Optimization Problems" in the mathematical section "Linear Programming." The developed program can be used by students from various training directions in universities and is intended to form deep theoretical knowledge and sustainable skills in solving specific problems. The program also includes testing to reinforce the covered material. It can be used in distance learning for part-time students and students combining work with education.

Thus, the development of computer-based educational programs is a relevant direction and is used by many authors in teaching various disciplines. The following section discusses issues related to creating computer-based educational programs and the requirements placed on them.

III. COMPUTER-BASED EDUCATIONAL PROGRAMS

Educational software is a term that refers to any computer software created for educational purposes. The scope of application for such software is not limited, as it is commonly used in the study of exact sciences as well as in linguistic analysis.

Computer-based educational programs, as software for educational purposes, can be conceptualized as a system consisting of two main components (subsystems): the informational (content) and the programmatic (implementation) [12].

The informational subsystem is responsible for the content of education, providing educational materials, concepts, and tasks. This part ensures the relevance, quality, and relevance of the instructional material.

The programmatic subsystem, in turn, is responsible for the technical implementation and presentation of the informational part. This includes interface development, task creation, management of the learning process, and other technological aspects.

When creating computer-based educational programs, there are two main approaches, represented by two polar points of view [12]. According to the first perspective, it is sufficient for the instructor to prepare educational materials correctly, and their computer adaptation will not present significant difficulties. According to the second perspective, a qualified programmer can take any textbook and, without the author's involvement, create an effective educational tool. The first approach emphasizes the importance of content, while the second emphasizes programmatic implementation.

These two aspects, informational and programmatic, are interrelated and require attention and efforts from both content experts and programmers to create comprehensive and effective educational software.

Computer programs used in education should primarily possess certain basic characteristics. Research on and identification of such characteristics for linguistic programs were undertaken by V.G. Sibirtseva and N.Kh. Frolova in their work [13]. The following characteristics were proposed for evaluating educational software:

- Presence of a large volume of data;
- Free availability;
- User-friendliness;
- Multilingualism;
- Multifunctionality.

The authors [13] compiled a series of educational programs possessing these properties, such as TextAnalyst SDK, LF Aligner, LEKTA, and AntConc. Within the research, a survey was conducted, asking respondents to mark the characteristics that were important to them (see table 1).

Thus, when it comes to computer-based educational programs, there are several key requirements that help ensure their effectiveness [14]:

- Goal-oriented educational work, assuming that after completing the course provided by the computer-based educational program, the user should achieve specific goals in terms of acquired knowledge, skills, and abilities.

- Proper selection and structuring of educational material, implying that a critical factor in education is the correct choice and presentation of educational material. The program should structure the material in such a way that the learner receives only the necessary information presented in an optimal form for assimilation.

- Additionally, it is important to consider that users may have different levels of experience in working with computers. Therefore, the program interface should be accessible and intuitively understandable even for those who do not have advanced computer skills.

- An orientation towards the average learner is also necessary. Except for narrowly specialized programs, educational material should be adapted to be understandable for those with basic knowledge in the field.

- The program should provide educational materials and an interface that allow the user to independently absorb information without the constant intervention of an instructor.



Table 1 – Characteristics of Investigated Programs in Comparison [13].

Program	Great text volume	Free access	User-friendliness	Multilingual	Multifunctioning
TextAnalyst SDK	+	+	-	-	-
LF Aligner	+	+	+	+	-
LEKTA	+	-	+	-	+
AntConc	+	+	-	+	+

IV. RESULTS

At the Department of Information Technologies at Turan University, a computer-based educational program is being developed for the course "Fundamentals of Computer Modeling." The program is intended for students enrolled in the educational programs "6B06101 - Information Systems" and "6B06103 - Intelligent Robotics," enabling them to model random patterns of any nature, ranging from random events to random flows.

The educational program consists of 6 laboratory works (fig. 1). The program interface is designed in three languages: Russian, Kazakh, and English (fig. 2).

Each laboratory work includes various methods for modeling random patterns and the most common laws of their distribution (fig. 2).

For example, in laboratory work No. 3, "Modeling Continuous Random Patterns," methods such as the inverse function method, Neumann's rejection method, limit theorem method, and composition method are discussed.

Among the frequently encountered distributions of continuous variables, normal, uniform, linear, gamma, and exponential distributions are highlighted, and these are also covered in laboratory work No. 3.

The results of laboratory works can be presented as numerical data (fig. 2) or as graphs (fig. 3). Numerical data can be saved in an Excel file format (fig. 4).

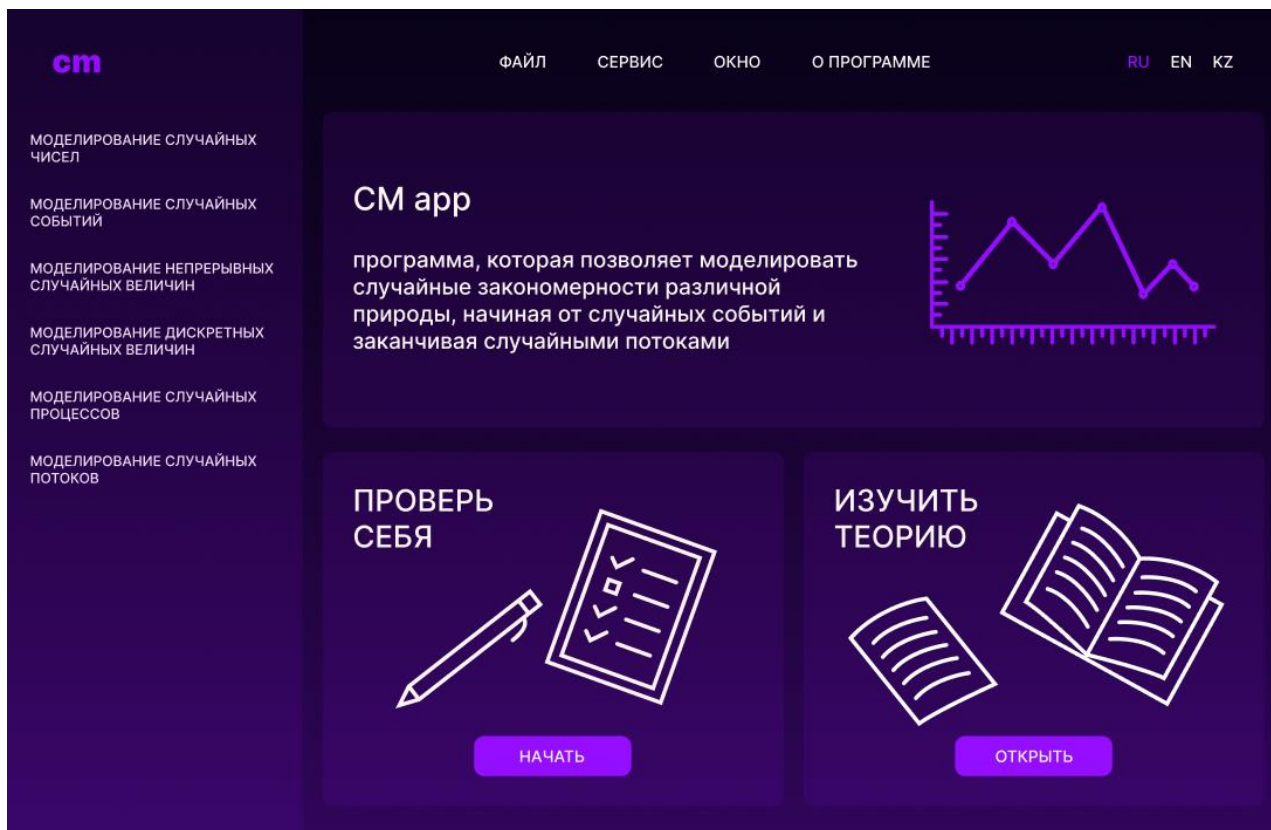


Fig 1 – Main Page of the Program

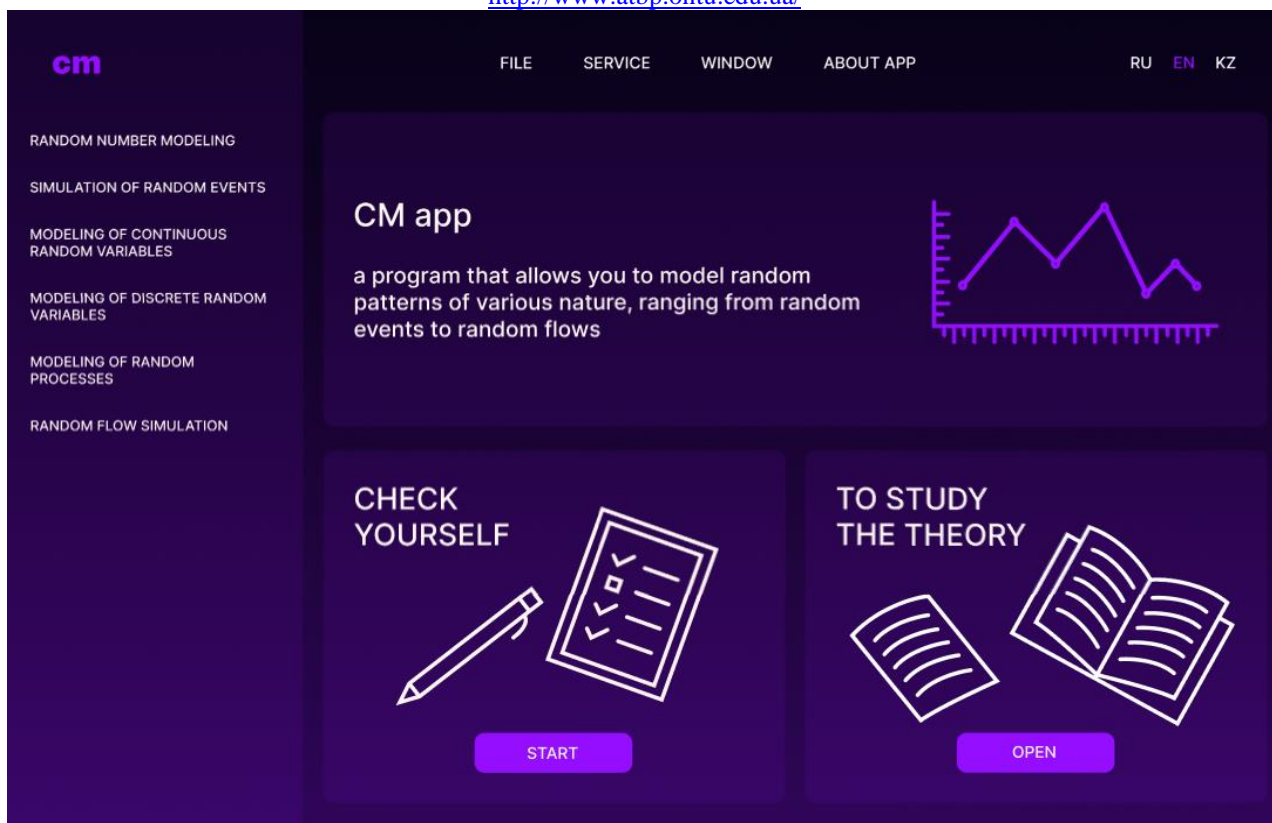


Fig 2 – The program interface is in English

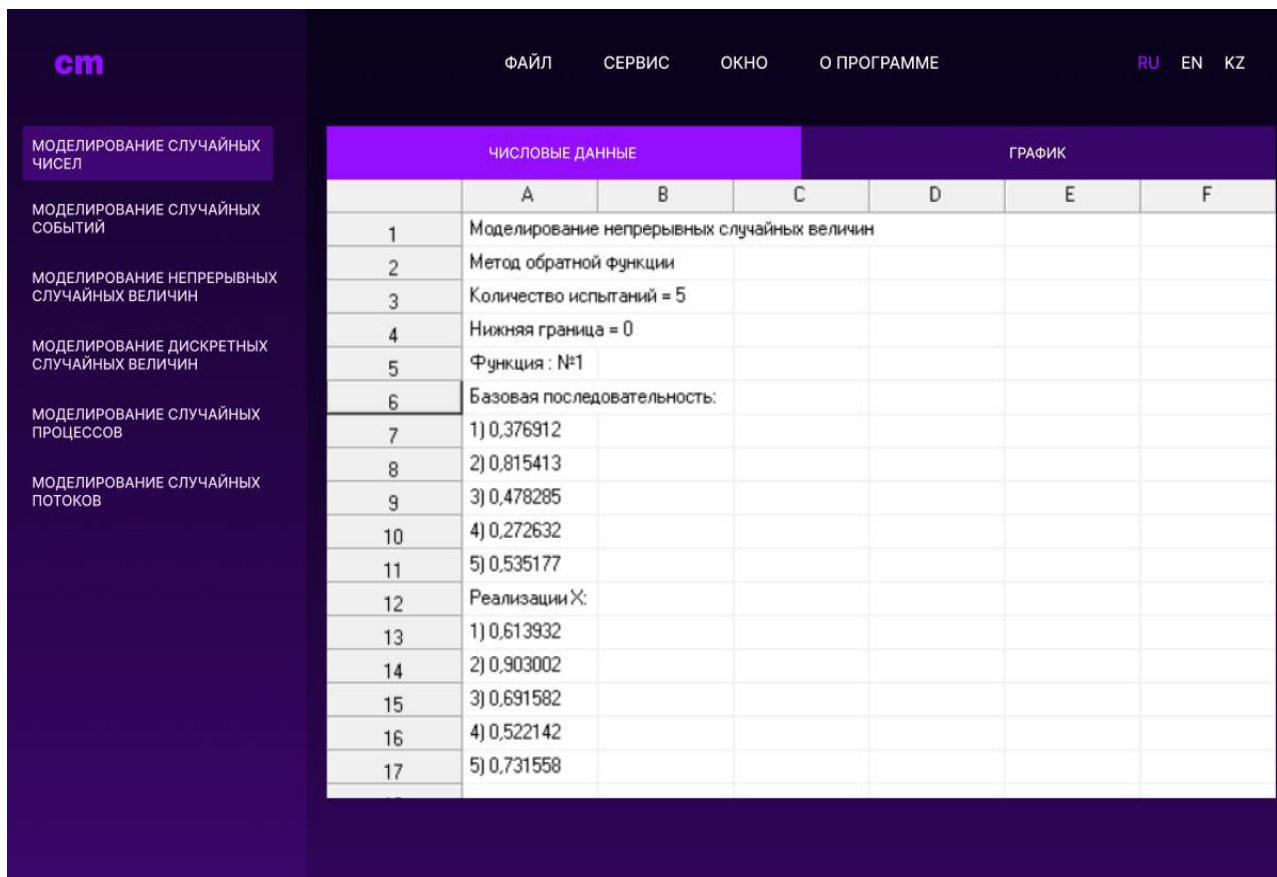


Fig. 3 – Modeling Continuous Random Variables

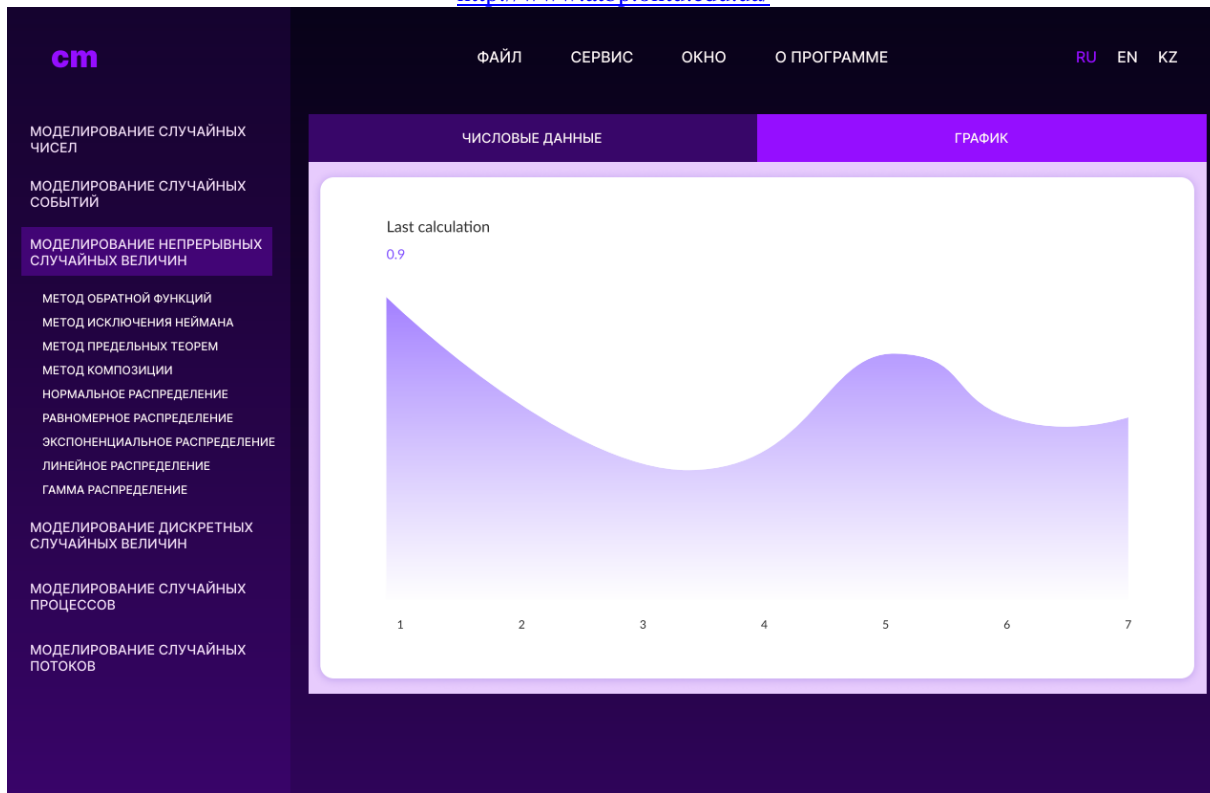


Fig.4 – Data output as a graph

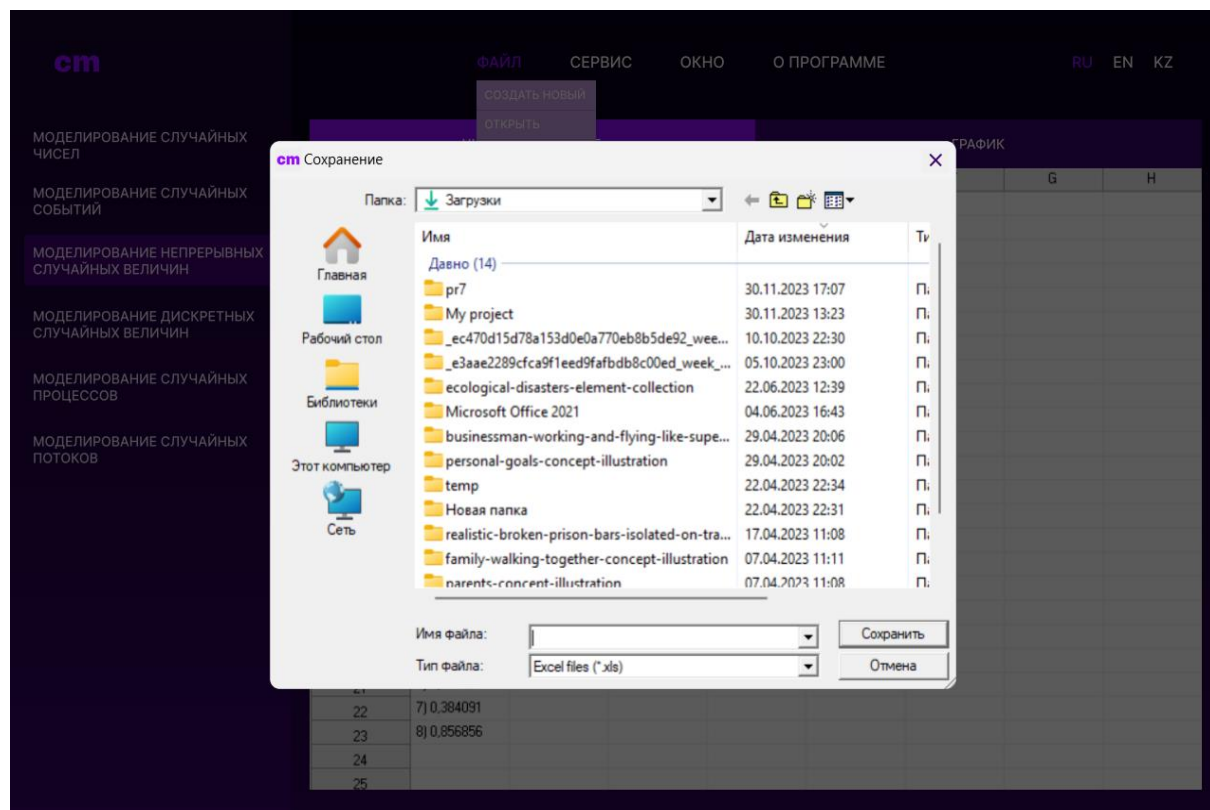


Fig.5 – Saving file as Excel format

Before completing each laboratory work, the learner can familiarize themselves with the theoretical part and numerical examples for each method and algorithm (fig. 6).

To reinforce the covered material, the learner can take a test on each topic and analyze their results (fig 7). In case of unsatisfactory results in a particular topic, the learner can review it again.



cm

ФАЙЛ СЕРВИС ОКНО О ПРОГРАММЕ RU EN KZ

МОДЕЛИРОВАНИЕ СЛУЧАЙНЫХ ЧИСЕЛ

МОДЕЛИРОВАНИЕ СЛУЧАЙНЫХ СОБЫТИЙ

МОДЕЛИРОВАНИЕ НЕПРЕРЫВНЫХ СЛУЧАЙНЫХ ВЕЛИЧИН

МОДЕЛИРОВАНИЕ ДИСКРЕТНЫХ СЛУЧАЙНЫХ ВЕЛИЧИН

МОДЕЛИРОВАНИЕ СЛУЧАЙНЫХ ПРОЦЕССОВ

МОДЕЛИРОВАНИЕ СЛУЧАЙНЫХ ПОТОКОВ

МОДУЛЬ 1. МОДЕЛИРОВАНИЕ СЛУЧАЙНЫХ ЗАКОНОМЕРНОСТЕЙ

Тема 1. Основные понятия модели и моделирования.

Система как объект моделирования

Объект – предмет, процесс или явление материального мира.
Декомпозиция объекта – разбиение объекта на части по определённым критериям.
Система – объект, рассматриваемый и как единое целое, и как совокупность взаимосвязанных элементов.
Системы классифицируются по многим признакам, среди которых пока отметим физические системы и абстрактные. Первые из них являются реальными системами, а вторые – отображениями (моделями) реальных объектов. Для реальной системы можно построить множество моделей, которые будут отображать те или иные её стороны или степень детализации.
Элемент – неделимая часть системы, представляющая единое целое.
Сложная (большая) система – система, содержащая большое количество элементов и связей между ними.
Комплекс – объединение связанных между собой систем.
Часто термины система и комплекс считаются эквивалентными.
Подсистема – функционально независимая часть системы, взаимодействующая с другими подсистемами.
Внешняя среда – объекты внешнего мира, воздействующие на функционирования системы, но не входящих в её состав.

ДАЛЕЕ

Fig. 6 – Theoretical part

cm

ФАЙЛ СЕРВИС ОКНО О ПРОГРАММЕ RU EN KZ

МОДЕЛИРОВАНИЕ СЛУЧАЙНЫХ ЧИСЕЛ

МОДЕЛИРОВАНИЕ СЛУЧАЙНЫХ СОБЫТИЙ

МОДЕЛИРОВАНИЕ НЕПРЕРЫВНЫХ СЛУЧАЙНЫХ ВЕЛИЧИН

МОДЕЛИРОВАНИЕ ДИСКРЕТНЫХ СЛУЧАЙНЫХ ВЕЛИЧИН

МОДЕЛИРОВАНИЕ СЛУЧАЙНЫХ ПРОЦЕССОВ

МОДЕЛИРОВАНИЕ СЛУЧАЙНЫХ ПОТОКОВ

ОСНОВНЫЕ ПОНЯТИЯ МОДЕЛИ И МОДЕЛИРОВАНИЯ

Вопрос 1

Объекты внешнего мира, воздействующие на функционирования системы, но не входящих в её состав, называется

- системой
- декомпозицией объекта
- сложной системой
- внешней средой
- подсистемой

Вопрос 2

Объект – это

- неделимая часть системы, представляющая единое целое
- множество элементов системы и связей между ними
- предмет, процесс или явление материального мира
- поведение системы, приводящее к результату под воздействием внешней среды и целями самой системы
- объединение связанных между собой систем

ЗАВЕРШИТЬ

Fig. 7 – Taking the test

V. CONCLUSIONS

In the context of rapid developments in information technology, the development of computer-based educational programs becomes one of the promising directions for the educational environment of any university. With the aim of enhancing the educational process and increasing the effectiveness of learning, it is proposed to integrate digital technologies into education and develop computer-based educational programs.

In this research work, the authors present a computer-based educational program based on a complex of simulation



models for modeling random patterns, demonstrating high universality. The proposed simulation system incorporates methods and algorithms for modeling random patterns of various natures, ranging from random events to random flows. These methods and algorithms cover the most commonly used methods and encountered distributions.

The developed computer-based educational program covers the entire cycle of the learning process, starting from studying the theoretical part, reinforcing it through practical examples, and concluding with testing. This program is designed to improve the learning process for the discipline and provides learners with the opportunity to independently study the subject without the direct intervention of an educator.

VI. REFERENCES

- [1.] Bogdanov S.G. Komp'yuternye obuchayushchie programmy kak osnovnoj ehlement informacionno-obrazovatel'noj sredy v organizacii processa podgotovki bakalavrov-kinologov / S. G. Bogdanov // Gumanizaciya obrazovaniya. – 2023. – № 2. – S. 132-143. – DOI: 10.24412/1029-3388-2023-2-132-143 (data obrashcheniya 25.01.2024).
- [2.] Chistyakov V.A. Komp'yuternye obuchayushchie programmy: problemy organizacii uchebnogo processa // Nauchnyj poisk. – 2014. – № 2.2. – S. 49-51. – URL: <https://elibrary.ru/item.asp?id=21843471> (data obrashcheniya 25.01.2024).
- [3.] Andreeva, E. A. Komp'yuternye obuchayushchie programmy po inostrannomu yazyku kak forma realizacii kompetentnostnogo podkhoda v vuze / E. A. Andreeva, R. R. Sagitova, EH. I. Nazmieva // Vestnik NCBZHD. – 2019. – № 1(39). – S. 5-9. – URL: <https://elibrary.ru/item.asp?id=37197030> (data obrashcheniya 25.01.2024).
- [4.] Blinov S.N., Ivanilova T.N., Kudryavceva O.A. Obuchayushchaya komp'yuternaya programma "Virtual'nye laboratornye raboty po kursu "Fizika" dlya studentov tekhnicheskikh special'nostej" // Otkrytoe i distancionnoe obrazovanie. – 2019. – № 1(73). – S. 46-52. – DOI 10.17223/16095944/73/6.
- [5.] Safonova I.E., Shcherbakova O.S. Komp'yuternaya obuchayushchaya programma "Issledovanie modelej SMO i SEMO" // Informacionnye tekhnologii v nauke, obrazovanii i proizvodstve: Materialy mezhdunarodnoj nauchno-prakticheskoy konferencii, Moskva, 08 fevralya 2018 goda / Pod red. YU.S. Rudenko, R.M. Kubovoj, V.V. Shamraevoj. – Moskva: Moskovskij universitet im. S.YU. Vitte, 2018. – S. 323-336.
- [6.] Pozanova I.A. Ispol'zovanie ehlektronnykh sredstv obucheniya na zanyatiyakh po discipline "Kriminalisticheskoe issledovanie veshchestv, materialov i izdelij iz nikh" / I. A. Pozanova // Vestnik Vostochno-Sibirskogo instituta MVD Rossii. – 2012. – № 3(62). – S. 14-19.
- [7.] Kostenkov V.A., Maslov A.D., Utenkov A.YU. Osobennosti ispol'zovaniya komp'yuternykh obuchayushchikh programm // DINAMIKA VZAIMOOTNOSHENIJ RAZLICHNYKH OBLASTEJ nauki v SOVREMENNYKH USLOVIYAKH: sbornik statej Mezhdunarodnoj nauchno-prakticheskoy konferencii, Novosibirsk, 29 maya 2018 goda. Tom Chast' 1. – Novosibirsk: Obshchestvo s ogranichennoj otvetstvennost'yu "OMEGA SAJNS", 2018. – S. 32-34. – URL: <https://elibrary.ru/item.asp?id=34936337> (data obrashcheniya 25.01.2024).
- [8.] Kostenkov V.A., Maslov A.D., Volkov A.A. Osobennosti ispol'zovaniya testovykh programm i laboratornykh praktikumov // DINAMIKA VZAIMOOTNOSHENIJ RAZLICHNYKH OBLASTEJ nauki v SOVREMENNYKH USLOVIYAKH: sbornik statej Mezhdunarodnoj nauchno-prakticheskoy konferencii, Novosibirsk, 29 maya 2018 goda. Tom Chast' 1. – Novosibirsk: Obshchestvo s ogranichennoj otvetstvennost'yu "OMEGA SAJNS", 2018. – S. 35-37. – URL: <https://elibrary.ru/item.asp?id=34936337> (data obrashcheniya 25.01.2024).
- [9.] Blinov S.N., Orlov A.A. Obuchayushchaya komp'yuternaya programma "Virtual'nye laboratornye raboty po discipline "Teplosnabzhenie predpriyatij lesnogo kompleksa" // Informatika i obrazovanie. – 2020. – № 10(319). – S. 54-61. – DOI 10.32517/0234-0453-2020-35-10-54-61.
- [10.] Shcherbatykh V.E. O primenenii komp'yuternoj obuchayushchej programmy pri izuchenii graficheskogo metoda resheniya zadach linejnogo programmirovaniya // Fundamental'nye problemy obucheniya matematike, informatike i informatizacii obrazovaniya: Sbornik tezisov dokladov mezhdunarodnoj nauchnoj konferencii, Elec, 29 sentyabrya – 01 2023 goda. – Elec: Eleckij gosudarstvennyj universitet im. I.A. Bunina, 2023. – S. 233-236. – URL: <https://elibrary.ru/item.asp?id=54642271> (data obrashcheniya: 26.01.2024).
- [11.] Sidorenko T.EH. Obuchayushchaya komp'yuternaya programma po inostrannomu yazyku: funkcii i sodержanie // Nauka segodnya: zadachi i puti ikh resheniya : materialy mezhdunarodnoj nauchno-prakticheskoy konferencii, Vologda, 29 maya 2019 goda. – Vologda: OOO «MarkeR», 2019. – S. 111-113. – URL: <https://elibrary.ru/item.asp?id=37750519> (data obrashcheniya: 26.01.2024).
- [12.] Ayazbaev T.L., Galaguzova T.A. Tekhnologiya sozdaniya komp'yuternykh obuchayushchikh programm // Mezhdunarodnyj zhurnal ehksperimental'nogo obrazovaniya. – 2015. – № 3-1. – S. 76-78. – URL: <https://expeducation.ru/ru/article/view?id=6702> (data obrashcheniya: 26.01.2024).
- [13.] Sibirceva Vera Grigor'evna, Frolova Natal'ya Khajdarovna Issledovanie potentsiala komp'yuternykh programm dlya stilisticheskogo i perevodcheskogo analiza teksta i ego prakticheskoe primenenie // Vestnik NGU. Seriya: Lingvistika i mezhkul'turnaya kommunikaciya. 2019. №1. URL: <https://cyberleninka.ru/article/n/issledovanie-potentsiala-kompyuternykh-programm-dlya-stilisticheskogo-i-perevodcheskogo-analiza-teksta-i-ego-prakticheskoe-primenenie> (data obrashcheniya: 26.01.2024).
- [14.] Nikol'skij P.G., Kuznecov A.V., Arkhipov S.N., Chistyakov S.V. Komp'yuternaya obuchayushchaya programma dlya podgotovki specialistov v oblasti telekommunikacij // Nauchnyj rezul'tat. Informacionnye tekhnologii. – 2019. – T. 4, № 2. – S. 53-62. – DOI 10.18413/2518-1092-2019-4-2-0-6.

Отримана в редакції 01.05.2024. Прийнята до друку 15.05.2024. Received 01 May 2024. Approved 15 May 2024. Available in Internet 30 July 2024