Ministry of Education and Science of Ukraine
Ternopil Ivan Puluj National Technical University

Faculty of Computer Information System and Software Engineering
(full name of faculty)
Department of Computer Science
(full name of department)

QUALIFYING PAPER

For the degree of

Bachelor
(degree name)

topic: Interactive learning information system

Submitted by: student IV course, group ICH-43
specialty 122 Computer science
(шифр і назва спеціальності)

(signature) Masope-Crabbe Festus (surname and initials)

Supervisor (signature) Holotenko O.S. (surname and initials)

Standards verified by (signature) (surname and initials)

Head of Department (signature) Bodnarchuk I.O. (surname and initials)

Reviewer (signature) (surname and initials)

Ternopil
2024

Ministry of Education and Science of Ukraine
Ternopil Ivan Puluj National Technical University

Faculty Faculty of Computer Information System and Software Engineering
(full name of faculty)

Department Department of Computer Science
(full name of department)

APPROVED BY
Head of Department

_____ Bodnarchuk I.O.
(signature) (surname and initials)

« » 20___

ASSIGNMENT
for QUALIFYING PAPER

for the degree of Bachelor
(degree name)

specialty 122 Computer science
(code and name of the specialty)

student Masope-Crabbe Festus
(surname, name, patronymic)

1. Paper topic Interactive learning information system

Paper supervisor Holotenko O.S., PhD
(surname, name, patronymic, scientific degree, academic rank)

Approved by university order as of «20» 12 2024 № 4/7-1202

2. Student's paper submission deadline 20.01.2025

3. Initial data for the paper Literature sources about architecture, principles of operation and development of information systems.

4. Paper contents (list of issues to be developed)

5. List of graphic material (with exact number of required drawings, slides)

6. Advisors of paper chapters

Chapter	Advisor's surname, initials and position	Signature, date	
		assignment was given by	assignment was received by
Life safety,			
basics of labor protection			

7. Date of receiving the assignment 20.01.2025

TIME SCHEDULE

LN	Paper stages	Paper stages deadlines	Notes
1	Analysis of the task for qualifying work. Selection		*Completed*
	and work with literary sources.		
2	Writing chapter 1		*Completed*
3	Writing chapter 2		*Completed*
4	Writing chapter 3		*Completed*
5	Writing chapter 4		*Completed*
6	Standartization control		*Completed*
7	Plagiarism check		*Completed*
8	Preliminary defense of qualifying paper		*Completed*
9	Defense of qualifying paper		

Student _____ Masope-Crabbe Festus
(signature) (surname and initials)

Paper supervisor _____ Holotebko O.S.
(signature) (surname and initials)

ANNOTATION

Interactive learning information system // Diploma thesis Bachelor degree // Masope-Crabbe Festus // Ternopil’ Ivan Puluj National Technical University, Faculty of Computer Information System and Software Engineering, Department of Computer Science // Ternopil', 2025 // P. __, Fig. – __, Tables – __, Annexes – __, References – __.

Keywords: learning, application, network, information system.

During the course of the thesis, a fully functional software for generating crosswords was obtained, which runs on the Windows operating system. The interface of the created program is convenient, simple, and intuitive improves its capabilities

The developed application is a tool for interactively creating crossword forms and filling them in. In other words, it is a crossword generator. Similar applications can be used by various publications that either specialize in publishing crosswords or sometimes decorate their products with them.

The developed software satisfies all the requirements set at the task formulation stage.

CONTENT

LIST OF ABBREVIATIONS, SYMBOLS AND TERMS

HEI - Higher education institution;

OS - Operating system;

ADT - Android Development Tools;

ALM - Application Lifecycle Management;

API - Application Programming Interfaces;

APK - Android Package;

ARM - Advanced RISC Machine;

ASP - Active Server Pages;

CPU - Central Processing Unit;

DSL - Domain-Specific Language;

IDE - Integrated Development Environment;

JDT - Java Development Tools;

MIPS - Microprocessor without Interlocked Pipeline Stages;

MVC - Model-View Controller;

OHA - Open Handset Alliance;

PDE - Plug-in Development Environment;

RCP - Rich Client Platform;

RISC - Reduced Instruction Set Computing;

SCM - Supply Chain Management;

SDK - Software Development Kit;

SE – Standard Edition;

SOA - Service-Oriented Architecture;

SSDT - SQL Server Data Tools;

SQL - Structured Query Language;

TFVC - Team Foundation Version Control;

XML - Extensible Markup Language;

INTRODUCTION

The first mobile device appeared in 1973 and quickly gained popularity all over the world. It was a means for making calls and talking. Nowadays, mobile devices are an integral part of the life of a modern person. They are multifunctional: they allow people to communicate, perform the functions of a photo and video camera, give the opportunity to listen to music, use Internet resources, play games and much more.

Nowadays, the most widespread and accessible OS is Android. The main advantages of Android OS are that it supports most smartphones and provides free development tools (while iOS OS requires considerable costs).

The second part of developing a thesis is creating a crossword puzzle.

The crossword puzzle appeared at the beginning of the 20th century.

The word crossword is formed from two English words cross - intersection and word - word. Solving crosswords helps train memory, broadens horizons, and some types of crosswords contribute to the development of intelligence and associative thinking.

Crossword puzzles are used in various fields. For example, teachers also use crossword puzzles in their work, this allows students to learn knowledge in a playful way. The scope of crossword puzzles is constantly expanding, crossword puzzles are used to test professional qualities.

Compiling a crossword puzzle is a rather laborious and painstaking activity, requiring attention and perseverance. To complete crossword puzzles, you must have access to dictionaries with lists of words and definitions.

The topic of my diploma project is "Development of an information system for interactive learning tools."

The first software developed is an Android-based information system, a game - a quiz "Test Yourself", which is focused on checking the learned material.

after lectures. A quiz is a game aimed at a person's cognitive development, testing logic, memory, and thinking.

The second software being developed is software for the Windows operating system. It should provide the ability to automatically generate crosswords according to a user template, filling it with words from the dictionary.

1. ANALYSIS OF THE SUBJECT AREA AND STATEMENT OF THE TASK

1.1. Analysis of interactive methods, forms and means of learning

Interactive teaching methods today are an urgent way of working for a teacher in a classroom, a trainer in a group, a teacher in any educational institution. Unlike traditional ones, interactive teaching methods are based on the fact that participants will actively interact in the learning process and with each other.

Depending on the nature of the interaction between teacher and students, passive, active, and interactive teaching methods are distinguished.

Passive methods are learning in which the teacher is the active person who controls the course of the lesson (Fig. 1.1). Students play the role of passive listeners and obey the teacher's orders [1]).

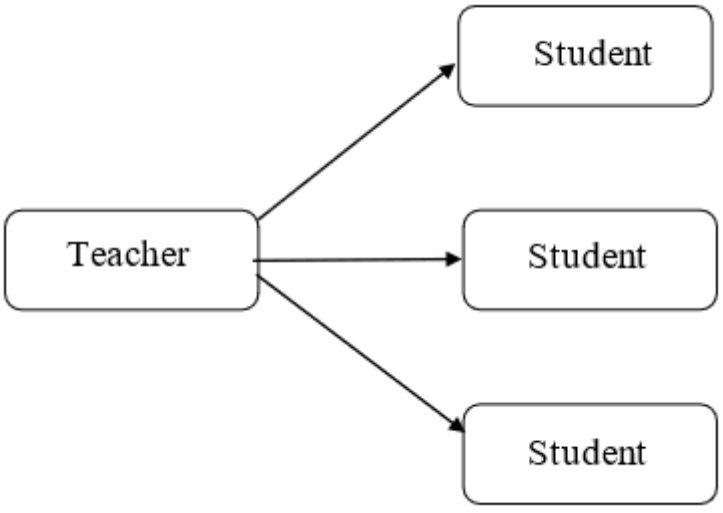

Figure 1.1 – Scheme of the passive learning method

Communication between the teacher and students is carried out through surveys, independent work, tests, and quizzes.

Considering the effectiveness of learning material, passive methods are not very effective. However, they have some advantages: it is relatively easy for the teacher to prepare for the lesson, there is an opportunity to teach a large amount of educational material in a limited time, and the opportunity to work with a large audience.

Active methods are learning in which students and the teacher interact with each other (Fig. 1.2). Here, students are not passive, but active participants. Passive methods have an authoritarian style. Active methods emphasize a democratic style.

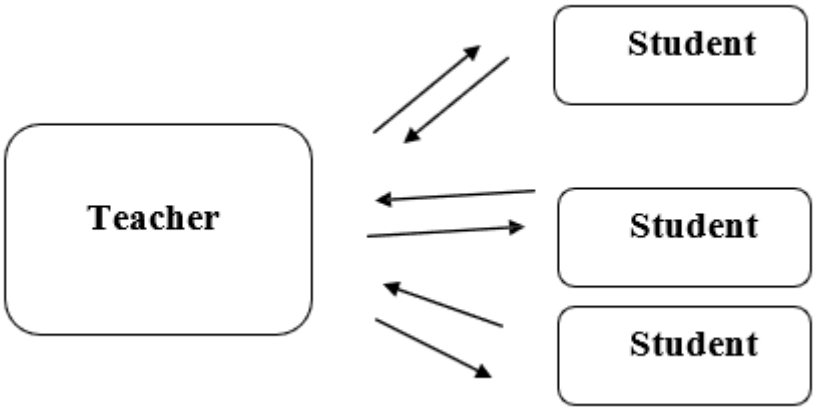

Figure 1.2 – Scheme of the active learning method

Interactive methods are a form of learning in which students and teachers interact with each other, conduct a dialogue with each other (Fig. 1.3). This is mutual learning: teacher - student, student - student. At the same time, the teacher and student are equal. This method excludes the dominance of one participant over the other.

Interactive methods, unlike active ones, are focused on the interaction of students not only with the teacher, but also with each other, on the activity of students in the process learning. The word "interactive" comes from the English

language and is derived from the word "interact". "Inter" means "mutual", and "act" means "to act".

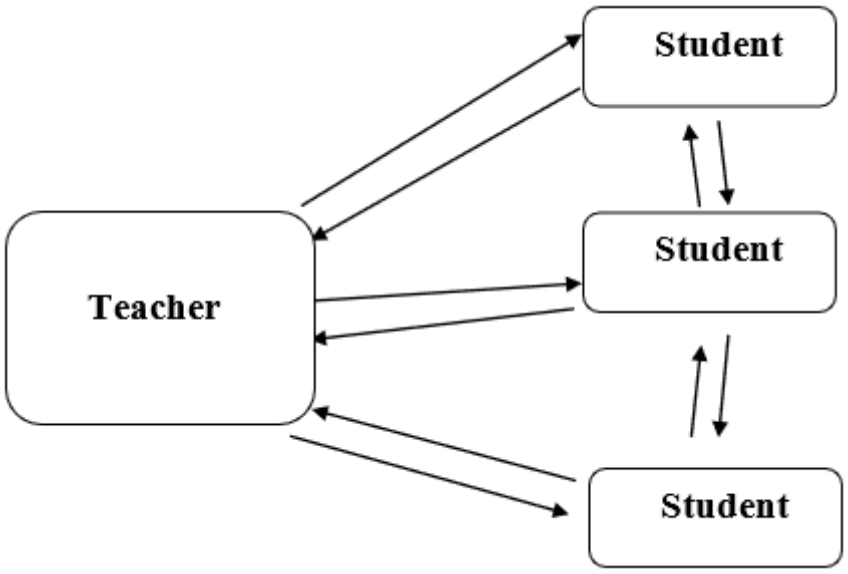

Figure 1.3 – Diagram of an interactive learning method.

Using interactive technologies, students learn:

– analyze educational information and approach the learning of the material creatively;

– formulate one's own opinion, express it, prove one's position and argue;

– respect the opinions of others;

– build constructive relationships in the group, avoid and resolve conflicts, and seek compromises;

– work independently, do creative work.

In the 1980s, the National Training Center (USA, Maryland) conducted studies showing that interactive learning dramatically increases the percentage of material mastery, as it affects not only the student's mind, but also his feelings.

The results of these studies are reflected in a diagram called the “Learning Pyramid” (Fig. 1.4).

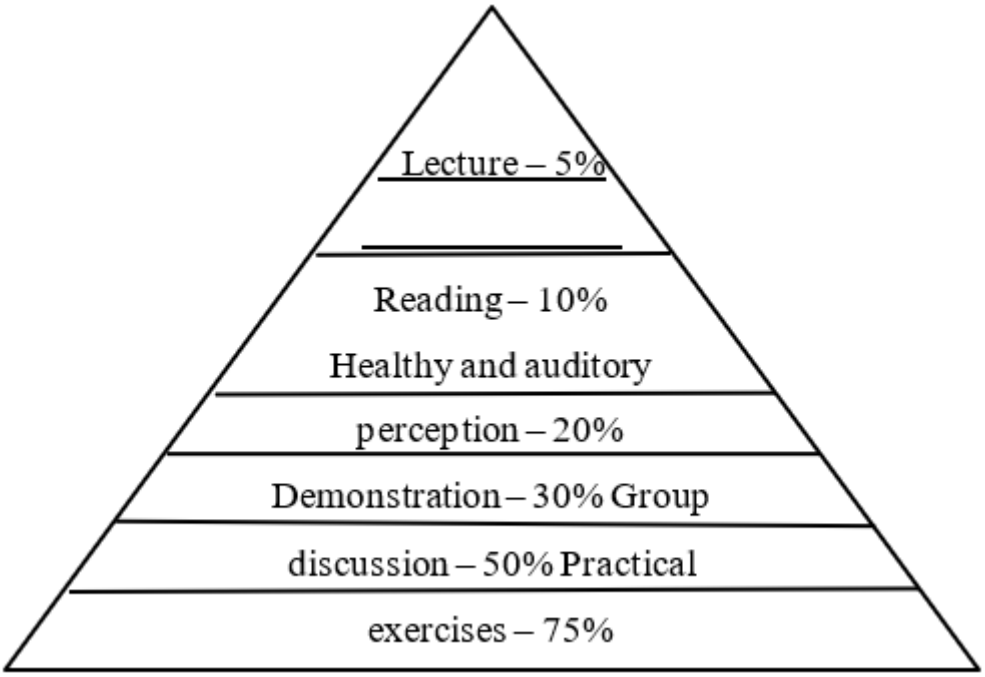

Figure 1.4 – Learning Pyramid

According to the pyramid, it is clear that the lowest results are achieved under conditions of passive learning: lecture – 5%, reading – 10%, visual and auditory perception – 20%, demonstration – 30%.[2]). And the highest results are achieved under conditions of interactive learning: group discussion – 50%, practical exercises – 75%, teaching others (practical) – 90%. These data are averages, so they may vary.

The following principles of interactive learning are distinguished:

- the principle of activity (each student must actively participate in the discussion process and actively interact with others);

- the principle of open feedback (group members express their thoughts, ideas, or objections), thanks to which they learn about the behavioral characteristics and thinking styles of others;

- the principle of experimentation (students actively search for new ideas and ways to solve the assigned tasks);

- the principle of trust in communication (usually the method of arranging the student and teacher in a circle facing each other);

- the principle of equality of positions (the teacher does not impose his opinion on the students, but on the contrary, acts together on an equal footing with them).

Types of interactive teaching methods: [3])

- Brainstorming method – students spontaneously put forward their ideas on a given topic. At the first stage, ideas are recorded, then they are classified and used for further analysis;

“round table” – listeners try to ask questions on the topic, argue methods for solving them;

- Discussion is an active method of conducting classes that mobilizes the participants' practical and theoretical knowledge, as well as their views on the problem being considered. Usually, questions are sought within the studied topic on which the students have different points of view;

- debate – an organized process of formulating and defending their positions on a specific issue by two or more participants;

- situational analysis – after learning about the problem, students independently analyze the situation and provide solution ideas to other students.

They distinguish:

- situations – illustrations (contains an example from management practice and a way to solve the situation), situations – assessments (a description of the situation and its possible solution is ready-made, an assessment of effectiveness is required), situations – exercises (a prepared episode of a specific exercise is prepared in such a way that its solution requires standard actions);

- case study analysis – a case study analysis that involves an in-depth and detailed study of a real or simulated situation.

1.2. The practice of using games in learning

Psychologists believe that play is one of the main types of human activity, and covers all periods of his life. The game form of learning is a way of interaction between the teacher and students, conditioned by the game, which leads to the definition and implementation of didactic goals and learning objectives.

Usually, the approximation of the student's actions in the classroom to real professional activity has the character of a certain convention, which is a characteristic feature of game actions. Such modeling, aimed at the orientation of professional behavior, in the conditions of learning, occurs thanks to the teacher, who develops a complex of various game situations. The role of the teacher in this is to offer the student educational situations that have a professional orientation for the student.

It is in such simulation-game situations that students can associate themselves with the role of the person they are playing. They can also put themselves in the place of this person in certain circumstances that will help enrich their life experience. This becomes possible precisely when modeling virtual professional situations in professional training. Some researchers (W. Rivers) call such activities an imitation-modeling game. In practice, the differences between role-playing games and simulations are mostly conditional, since the goal of both simulations and role-playing games is to form students' professional skills while solving various professional problems.

In this way, the role-playing behavior of future social workers is realized in the conditions set or created by the teacher.

In role-playing games, participants' views or beliefs may not coincide with those of their characters, while simulations allow participants to be themselves.

It is for this reason that many researchers distinguish between the concepts of role-playing and business games and simulations. In a role-playing game, the views or beliefs of participants may not coincide with the views of their characters,

while simulations allow participants to be themselves. Compared with real professional activities, when a decision or action leads to certain results, in game interaction it is possible to change the course and method of activity, try different options for professional behavior, and make predictions about its manifestation in the future. This allows future specialists to choose the most optimal and effective way to form their own professionalism and professional competence.

Play is a universal form in which powerful processes of self-determination, self-discovery, self-affirmation, and self-testing occur. Games develop intelligence and logic.

Therefore, among the most widely used means of interactive technologies used in higher education are game learning technologies. Since there are different types of games used in the educational process (business, role-playing, didactic, simulation, etc.), and each of these types has its own implementation technology and is determined by the process of preparation for its implementation (game design), we can talk about different game technologies.

An important feature of interactive games is their ability to motivate students' learning, promote their socialization and professional development. Therefore, the educational impact of interactive games allows the student to set standards by which he measures the degree of satisfaction with his own behavior.

Currently, many options for game exercises are offered for teaching schoolchildren and students, but the mechanical use of interactive games does not guarantee the success of the educational process.

Advantages of interactive games:

motivates learning (awakens students' curiosity about ways to solve professional and educational problems in a gaming environment);

promote personal development (create a lasting interest in self-development and unlocking one's potential);

facilitate the introduction of new communicative and behavioral norms;

encourage a better understanding and learning to use the full complexity of mental, social and organizational processes of communication between people;

The game provides students with the opportunity to solve complex problems without remaining passive observers;

reduce participants' anxiety, especially at the initial stage of work.

All game technologies for training specialists in higher education can be combined under the general concept of “didactic-business games”. They are simultaneously a form of organizing the educational environment, a means of training future specialists, a method of their professional training, which is implemented by the teacher according to a specially developed game scenario (Fig. 1.5) [4]).

The use of a didactic business game as a form of contextual learning is designed to solve the following pedagogical tasks:

stimulating students' cognitive motivation to ensure conditions for the manifestation of professional motivation;

development of theoretical and professional-practical thinking in the professional field of future social workers;

formation of a holistic understanding of students' future professional activities and their dynamics.

According to the game method, didactic games can be divided into two large groups: simulation (analysis of specific situations, logical thinking, creative and cognitive, which include "Brainstorming", "Brain Ring",

"Broken Telephone", etc.) and role-playing, which have a certain logical-content plot, professional-business or socio-psychological orientation, and drama.

Therefore, they combine organizational-activity, professional-communicative, situational-plot, dramatic games, which are distinguished as independent in other classifications, and simulation-game situations.

In the process of implementing interactive game technologies of both categories, subject means can be used as the main and auxiliary material (technical

teaching aids, forms, tables, handouts, markers, blackboard, colored chalk); methodological means of interactive technologies (methodological developments and scenarios of games and game exercises) and game design.

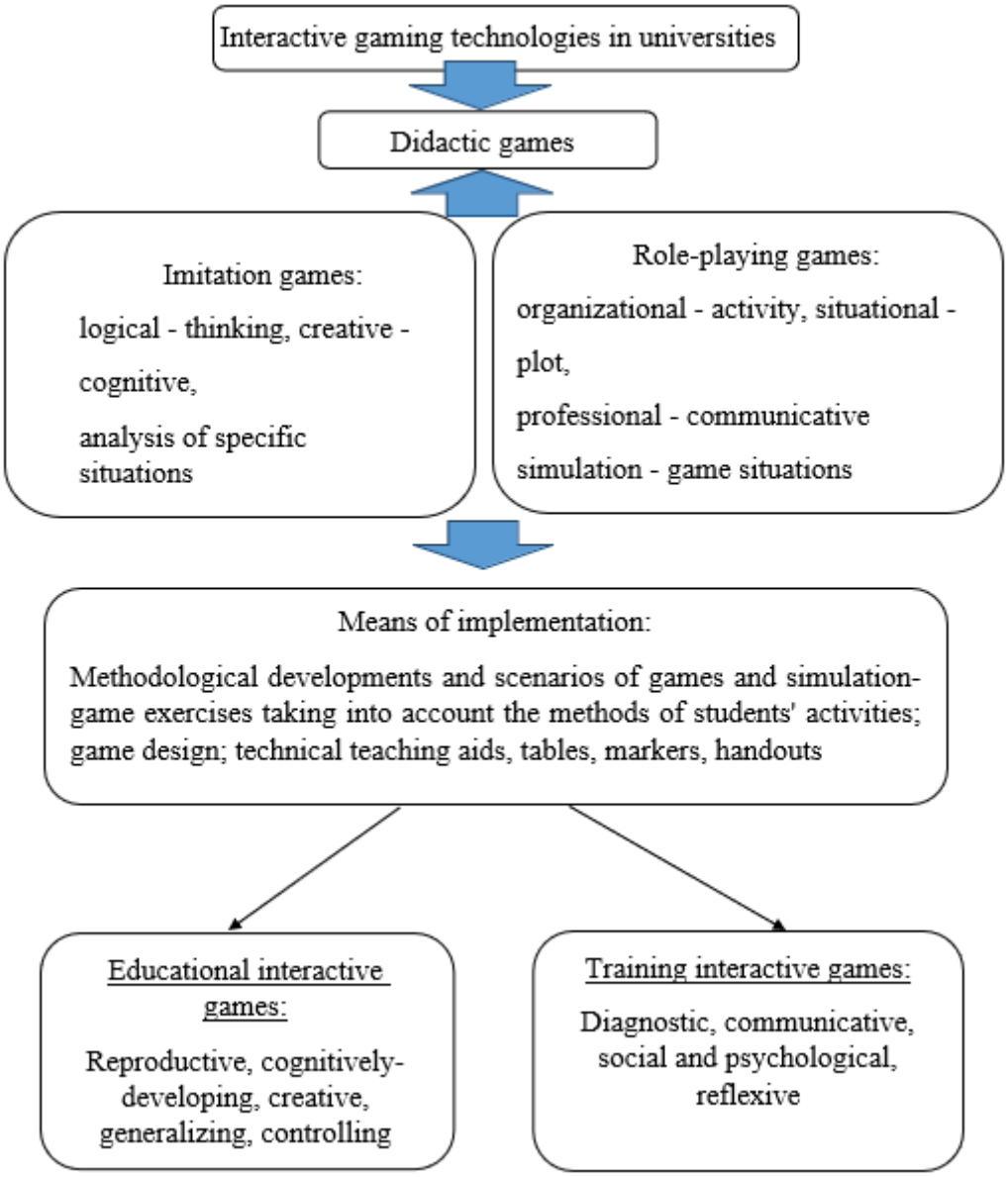

Figure 1.5 – Classification of interactive gaming technologies for higher education

Depending on the purpose, tasks and stage of application of imitation or role-playing games, they are characterized as educational (reproductive, cognitively and developmental, creative, generalizing, controlling) and training (in

the sense of working out and forming the necessary skills and abilities of activity), which include diagnostic, communicative, socio-psychological, and reflective games.

The methodology for applying interactive gaming technologies involves the implementation of the following main stages:

1) Preparing for a game-based lesson:

– contains a definition of the goal of the game and its time limit;

– complianceitscontent content of the academic discipline and topic of the lesson;

– developing your own or adapting a ready-made game script;

– minute-by-minute structuring of stages of game interaction (motivational, activity-based, reflective, control-evaluative);

– providing the necessary equipment and materials;

– development of clear step-by-step instructions for students - participants in game interaction (goal, tasks, rules, game actions, conditions).

2) Direct game implementation, which involves students' activities as a manifestation of their internal (psychological), external (physical) and social (professionally oriented) activity.

Activity is regulated by a conscious goal and has such characteristics as:

– prediction of the outcome;

– awareness of the possibility of achieving it;

– well-founded planning of one's actions;

– choosing appropriate rational forms, methods, means and guidelines in interpersonal relationships;

– harmonization of internal and external activities;

– evaluating the process and results of one's work, which requires the ability to standardize, take into account, control, make optimal decisions, solve general organizational tasks (diagnose, predict, stimulate activity, approach it

comprehensively, systematically, holistically) and enable students to check the feasibility of the activity and its proportionality to the set goal.

3) Analytical and evaluative stage, which involves reflection, generalization of game results, formulation of conclusions, determination by students of positive aspects of the game's effectiveness for their personal and professional growth in the future, and teachers' identification of shortcomings, omissions, and ways to improve students' game interaction for further use of the tested methodology in the future.

Gaming technologies can be considered as:

– a tool for translating and assimilating previous professional experience;

– analysis of reality models (on samples of professional actions by representatives of different role and personal positions);

– adaptation to future professional activities.

The use of didactic games contributes to the transformation of the student from

the object of learning into a subject of professional directed work, which causes his purposeful activity and creative participation in the independent formation professional competence.

Therefore, the analysis of the essence of interactive gaming technologies makes it possible to determine the main features of their application in the educational process of universities.

1.3. Crossword in the educational process

The most effective and convenient way to develop students' terminological literacy is a crossword puzzle.

Crossword is a game technique, the essence of which is to solve words according to the given definitions.

Each subject puts forward its own classification of crosswords, and this follows from the uniqueness of this educational discipline. Therefore, it is easier to take the main educational goal of the lesson as a basis.

Crossword helps:

organize independent work;

arouse interest in the topic being studied;

the formation of terminological literacy, that is, a more solid mastery of the terminological apparatus of the academic discipline [5].

When creating crossword puzzles, it is necessary to adhere to the principles of clarity and accessibility.

Crossword puzzle rules:

No empty cells are allowed in the crossword grid;

random letter combinations and intersections are not allowed;

the words guessed must be nouns in the nominative singular;

two-letter words must have two intersections;

three-letter words must have at least two intersections;

Abbreviations (ZIL and others), reductions (orphanage and others) are not allowed;

A large number of two-letter words is not recommended. Advantages of crossword puzzles as a learning method:

allows you to detail specific sections and complex topics of the academic discipline;

In a crossword puzzle, the guessed units (terms) must be unambiguous, concise, and specific. This ensures quick memorization of the term and its meaning;

Attention, memory, logical thinking, and language are activated. Forms of working with crosswords:

organization of independent extracurricular activities of students. This contributes to the development of the ability to independently and quickly navigate

the educational material. correctly and accurately formulate questions, determine the type of crossword puzzle and the need to apply it specifically to this topic, creating to solve crossword puzzles in printed and electronic form, and this in itself contributes to the development of a person's creative abilities;

organization of work in the lesson. At the stage of repeating the material: five-minute test tasks. At the stage of checking knowledge: working with terms and concepts of the academic discipline on a specific topic. At the stage of studying new material: predicting the future result.

Stages of compiling educational crossword puzzles:

we choose the type of crossword: it is better to use asymmetrical, non-standard, with free arrangement of words;

We make a list of terms (depending on the type of lesson and its goals).

Criteria for evaluating the result. Undoubtedly, the criteria depend on many factors and must be determined by the teacher individually, we can only highlight some basic points:

share guess the words;

share of key terms;

accuracy and clarity of question wording.

1.4. Analysis of existing interactive learning tools

In the process of completing the thesis, games of a similar type on the Google Play platform, as well as on the Internet, were considered. From the large list of applications, several successful ones can be distinguished.

The first information system is “Online Test Pad” (Fig. 1.6):

main page: contains all the necessary information about choosing a crossword topic or creating it;

it is possible to create crosswords, surveys or tests;

the service is absolutely free;

there is an opportunity to complete crossword puzzles, tests, and surveys online [6].

Brief description of the information system. The information system is very extensive, has many functions. Convenient interface, the content of the site is very full. There is the possibility of registration, login. You can chat. On the site you can create your own crosswords, surveys or tests, or use ready-made ones. There is the possibility of commenting and rating. On the main page there is a search function on the site, which makes it easier to work with it. There are also links to social networks.

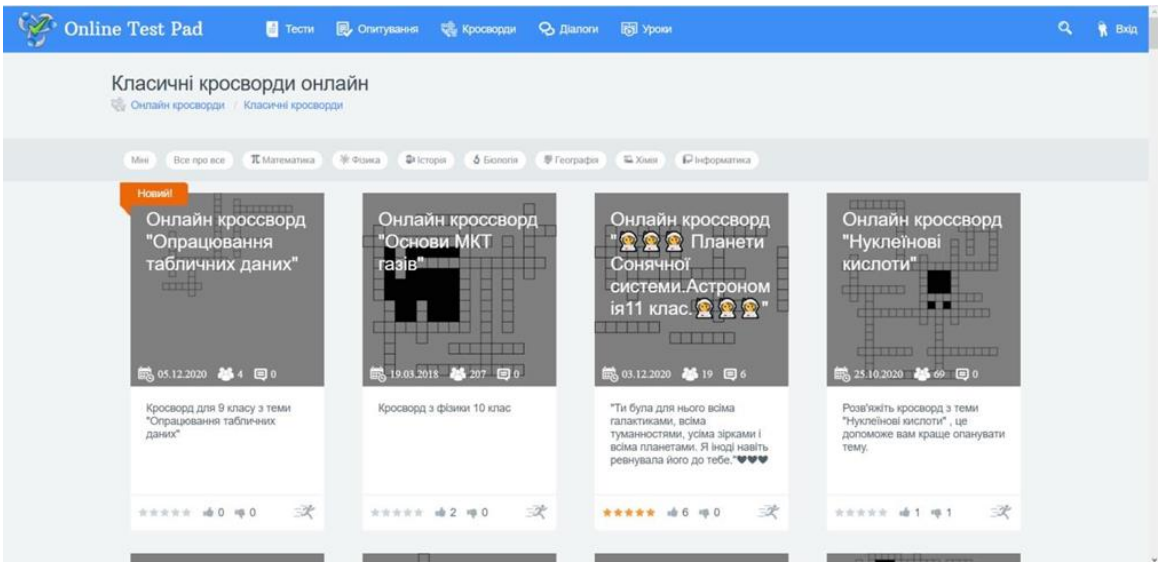

Figure 1.6 – Information system “Online Test Pad” The second information system “Source” (Fig. 1.7).

home: here you can find the main page with a description of the site, services, and contacts;

- separate sections for teachers and students;
- a separate page with answers to questions;
- registration option [7].

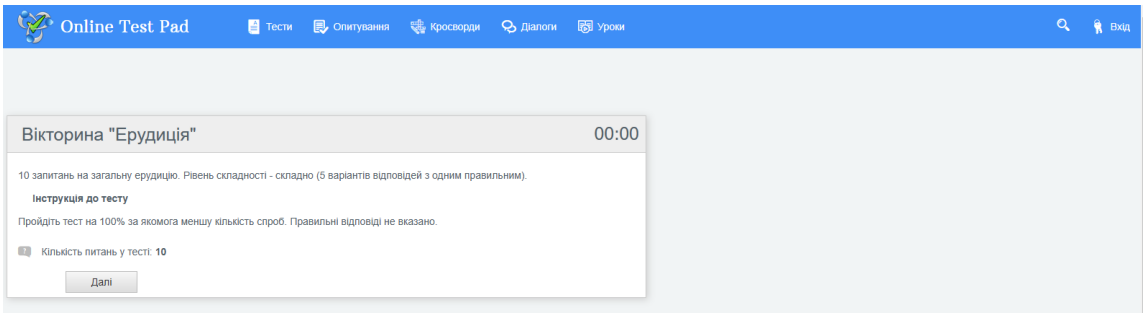

Figure 1.7 – Information system “Source”

Short description of the site: the design of the site has a rather pleasant lightness. The main page of the site includes all the necessary information. On the site

"Source" is, first of all, a lot of information. There is nothing superfluous on the main page. The list of offered programs is very compact, and due to this you can see as much information as possible at a time. It is very convenient and saves time. After registration, you have access to the participant's personal account, which allows you to receive test results in your mailbox and the opportunity to issue a diploma.

Let's consider and analyze the mobile application "World History Quiz" (Fig. 1.8).

A quiz game for anyone who wants to test their knowledge of world history from school.

Has a choice of number of questions. You can also choose the number of players (competition), and play for survival or time. The game has over 50,000 downloads.

Has 8 modes:

- 10 questions;
- 20 questions;
- 30 questions;
- 1 minute;

- 3 minutes;
- 5 minutes;
- for survival;
- for survival (3 attempts)[8].

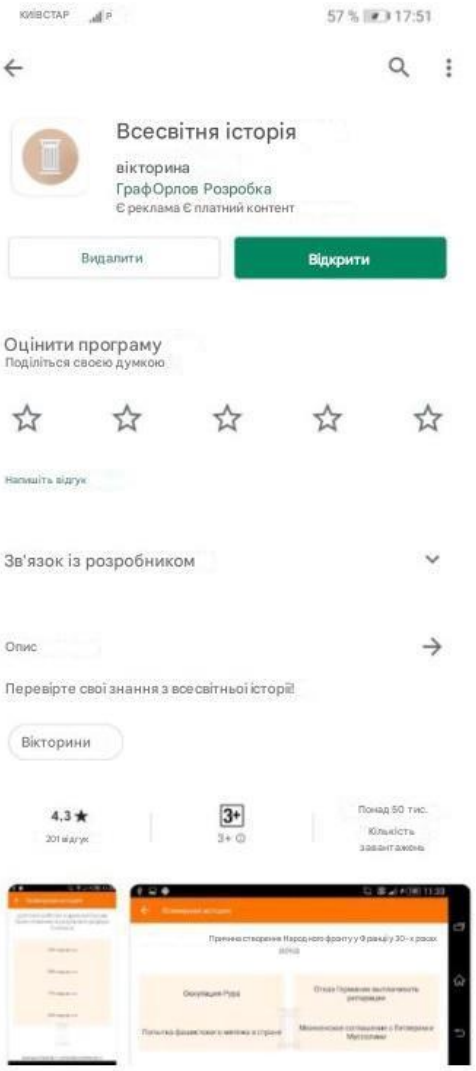

Figure 1.8 – Mobile application “World History Quiz”

Let's consider the following mobile application - "Crosswords" (Fig. 1.9).

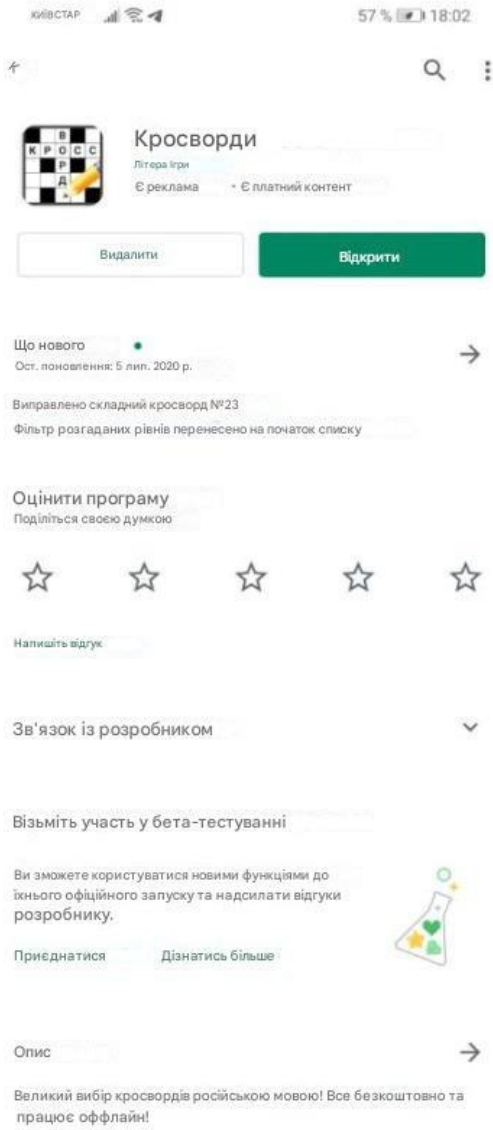

Figure 1.9 – Mobile application “Crosswords”

The game has more than 1,000,000 downloads. Short description of the application: more than three hundred crosswords. The application can work offline.

Categories to choose from:

– classic;
– thematic;
– all words with a given letter;
– curly;
– complex.

Key features:

– several different categories of crosswords;
– all crosswords can work offline;
– easy and difficult crosswords in terms of difficulty;
– clues for discovering letters and words [9]1).

1.5. Setting the task

Based on the conducted research of similar information systems, requirements for the designed system were formulated.

The goal of my thesis is to develop a mobile application "Quiz" based on the Android Studio platform and develop a crossword puzzle using the Visual Studio integrated development environment.

To achieve the goal, the following tasks were set:

– to formulate the problem;
– to review software tools for developing cross-platform mobile games;
– design applications;
– implement and test applications.

2. ANALYSIS OF EXISTING DEVELOPMENT TOOLS

Nowadays, there are a huge number of both paid and free tools for developing games and software products, ranging from simple libraries for well-known programming languages to large editors with extensive functionality.

To create a quality game, you need to use quality tools. That's why I decided to use ready-made tools for developing games, applications, and software products.

As application development environments have become very popular, the number of them on the market is increasing every day. In this section, we will analyze the most popular gaming platforms that have proven themselves on the market.

2.1. Android operating system

Android OS is an operating system for smartphones, tablet computers, e-books, digital players, "smart" wristwatches, game consoles, netbooks, smartbooks, Google glasses, televisions, automatic car control systems and other devices. The OS is based on the Linux kernel and Google's own implementation of the Java virtual machine. It was originally developed by Android Inc., which was bought by Google in 2005. Later, Google initiated the creation of the Open Handset Alliance (OHA), which is now engaged in support and further development of the platform.

Android allows you to create Java applications that control the device through Google-developed libraries. The Android Native Development Kit allows you to port (but not debug) libraries and application components written in C and other languages.

Each new version of Android was given a name that corresponds to the name of any dessert. Below is the statistics on the use of Android versions in April 2020 (Fig. 2.1).

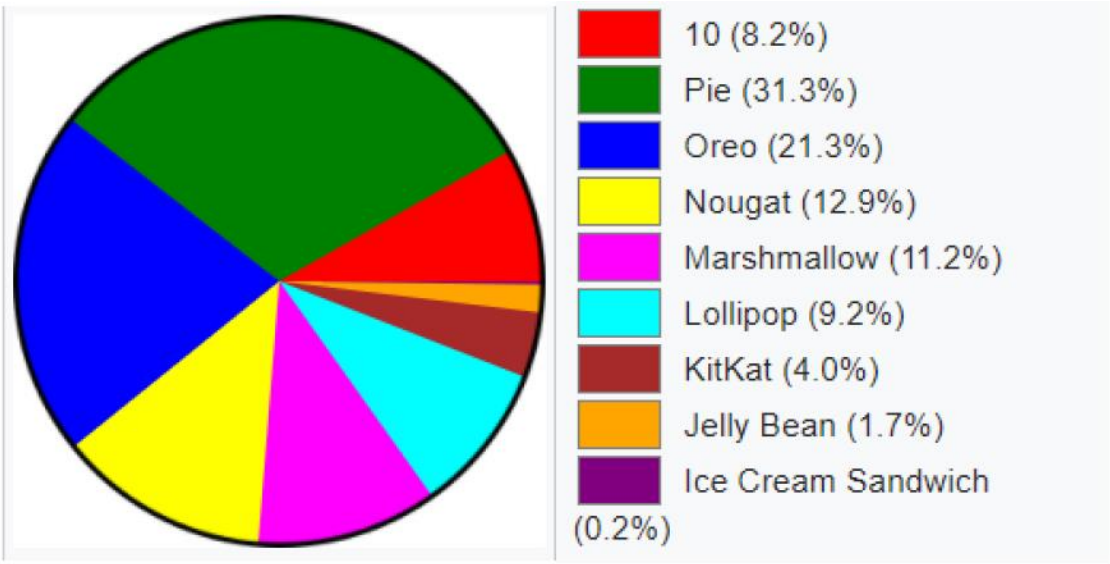

Figure 2.1 – Usage statistics for Android versions

The most popular areAndroid 9.0 "Pie",Android 8.0 / 8.1 "Oreo",Android 7.0 / 7.1 "Nougat"[10].

Android 9.0 "Pie" released on March 7, 2018. The kernel version is Linux, API version: 28. Key features:

– The RTT API allows you to measure the distance between a mobile device;

– innotifications display received images;

– users have the ability to block channel groups;

– updated appearance of the volume slider and date on the lock screen;

– a key to create a screenshot has been added to the power menu;

– a battery indicator has been added to Ambient Display;

– a single template for implementation has been introduceduser authentication by fingers crossed for everyone in the relationship.

Android 8.0/8.1 "Oreo" came out 21 August 2017 (8.0),5 December 2017 (8.1).

Kernel version is Linux: 4.10. API version: 26.27.

Key features of Android 8.0:

– regimepicture-in-picture;

– support for multiple displays;

– the notification system has been redesigned;

– the ability to postpone notifications for a certain time, or delete them;

– improved auto-completion of fields such as username and credit card number using the framework;

– pinning shortcuts and widgets inside any application. Key features of Android 8.1:

– Android Go is a simplified version of the Android distribution designed for low-end and ultra-budget smartphones;

– APIsneural networks;

– various security improvements and updates.

Android 7.0 / 7.1 "Nougat" came outAugust 22 2016 The kernel version is Linux: 3.10. API version: 24, 25.

Key features of Android 7.0:

– regime multi-window split screen, in which two applications can divide it in half;

– quick access icons are displayed on the compact panel;

– filtering incoming calls by phone number;

– background task switching;

– night mode allows you to achieve optimal display of information on the screen by automatically increasing contrast and adjusting brightness;

– Improved power saving feature "Doze". Previously it worked only when the phone was stationary, but in this versionGoogle claims that "Doze saves battery whenever the screen turns off";

– new folder design: icons inside the frame are arranged in a grid. Key features of Android 7.1:

– night mode "Night light" (blue filter);

– the ability to quickly respond to messages by swiping on the fingerprint scanner;

– automatic (background) security updates;

– Daydream VR support.

The Android architecture is formed from a set of components. Each component is built on the basis of lower-level elements (Figure 2.2). In the lower part of the figure, you can see that the Linux kernel provides basic drivers for the hardware components of the system.

In addition, the kernel is responsible for memory, process management, network support, etc. The Android runtime, which is a superstructure on top of the kernel, is responsible for generating and executing Android applications. Each application runs in its own process with its own Dalvik virtual machine. In addition to the kernel libraries, which offer some Java SE functionality, there is also a set of native C/C++ libraries that form the basis for the application framework [11].

These system libraries are mostly responsible for computationally intensive tasks (graphics rendering, sound playback, database access) that are not well suited for the Dalvik virtual machine. Their APIs are wrapped in Java classes in the application framework.

Application frameworks tie together system libraries and the runtime environment.

The framework manages applications and provides a well-designed environment in which they run.

Developers create applications for this framework using a set of Java APIs covering areas such as user interface development, background services, notifications, resource management, peripheral access, etc. All key applications that come with the Android OS (for example, the email client) are written using these APIs.

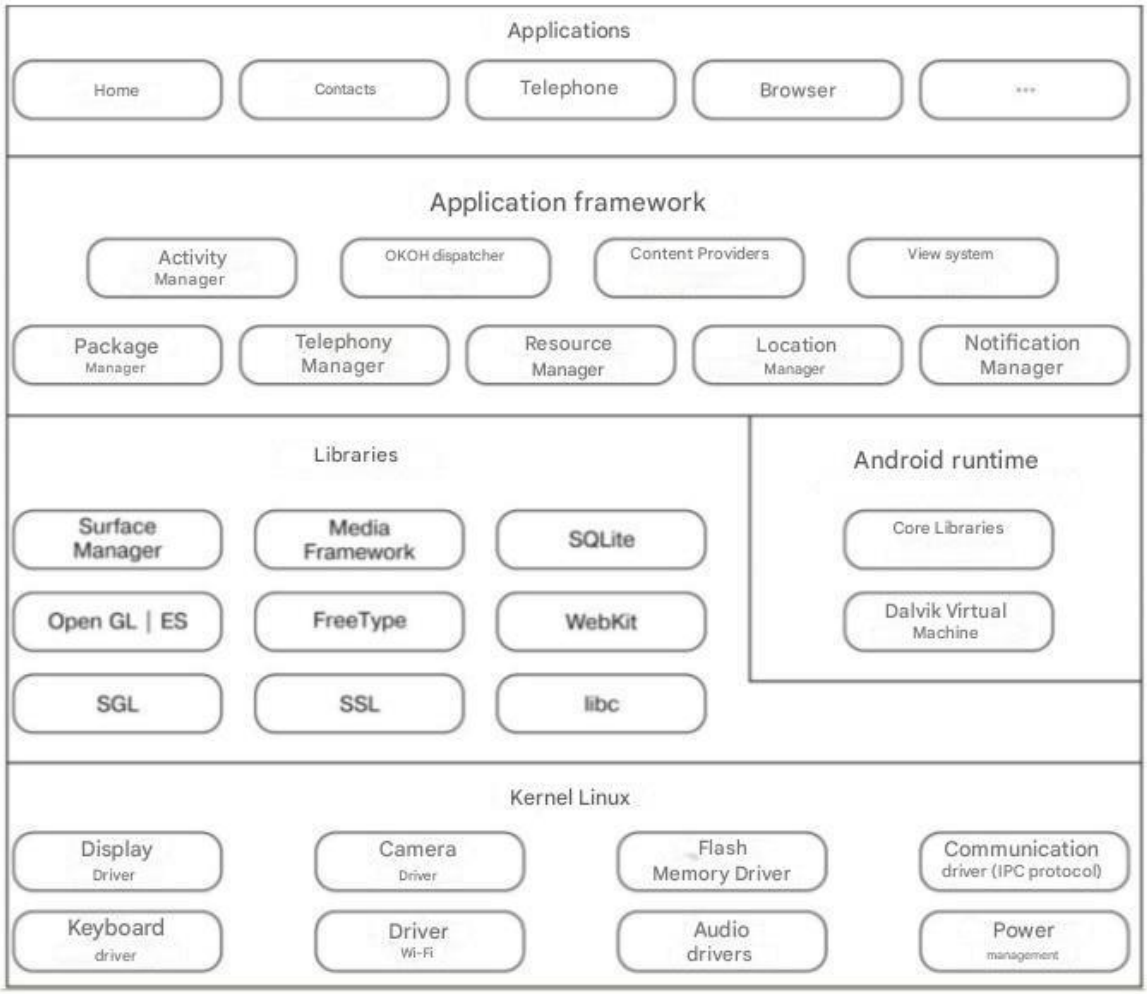

Figure 2.2 –Android OS Architecture

To develop game applications for the Android operating system, you can use the standard Android Studio application development tool. There are also several commercial and open source game engines and frameworks, such as Unity, Unreal engine, Torque2D, Cocos 2D-X, libGDX, Xamarin, Corona SDK.

Advantages of Android OS:

– Android, unlike iOS, is an open platform, which allows you to implement more functions on it;

– Android devices usually have a microSD card reader, which makes it possible to quickly transfer files from a computer to a phone;

– Despite the initial ban on installing applications from “unverified sources” (for example, from a memory card), this restriction can be disabled using

standard tools in the device settings, which allows applications to be installed on phones and tablets without an Internet connection (for example, users who do not have Wi-Fi).–access points and do not want to spend money on mobile Internet, which is usually expensive), and also allows anyone to write Android applications for free and test them on their device;

– Android is available for various hardware platforms such as ARM, MIPS, x86;

– There are alternative Google Play app stores: Amazon Appstore (English), Opera Mobile Store, Yandex.Store.

Disadvantages of Android OS:

– availability in some Android–Google services devices that provide the ability to transmit identification information to the company's servers, for example, information about the user's movement in real time.

– In Android version 1.6, developers added the Native Development Kit, which allows you to write your own low-level modules for the system in C /C++, based on Linux standards–libraries. Although, for example, the standard C language library on the Android platform, known as Bionic, is not standard and fully compatible with libc.

When working on this mobile application "Quiz", the Android Studio IDE was used. This tool will be considered below.

2.2. Software characteristics

2.2.1. Android Studio Integrated Environment

Android Studio is an integrated development environment (IDE) for working with the Android platform. It was announced on May 16, 2013 at the Google I/O conference (Figure 2.1).

The IDE has been freely available since version 0.1, released in May 2013, then entered beta testing starting with version 0.8, which was released in June 2014. The first stable version 1.0 was released in December 2014, at the same time that support for the Android Development Tools (ADT) plugin for Eclipse was discontinued.

Android Studio, based on IntelliJ IDEA software from JetBrains (the official Android application development tool).

This development environment is available for Windows, OS X, and Linux. New features appear with each new version of Android Studio.

The following features are currently available:

– working with layouts;

– editor object movement (MotionLayout). The MotionLayout API extends the capabilities of ConstraintLayout. Helps Android developers

control complex widget motion and animation. Using this API has become easier with the new Motion Editor - a powerful interface for creating, editing, and previewing.

Motion Editor creates and modifies complex XML files for us, while it supports editing transitions, keyframes, and view attributes. The editor automatically compiles the code;

– Layout Inspector. With the help of Layout Inspector makes UI debugging easier. It provides access to constantly updated data and information about how resources are being consumed;

– Layout Validation. When developing code for multiple screen sizes, the Layout Validation window allows you to see how layouts look on different configurations and screens at the same time, and how correctly they are displayed.

– development and profiling: CPU Profiler. Provides a large amount of information about thread activity and program tracing;

– Smart editor features: for creating files for R8 (which is introduced in Android Gradle plugin 3.4.0), the studio provides smart editor features: syntax highlighting, auto-completion and error checking. The editor integrates with the project, provides symbol completion of all classes, methods and fields;

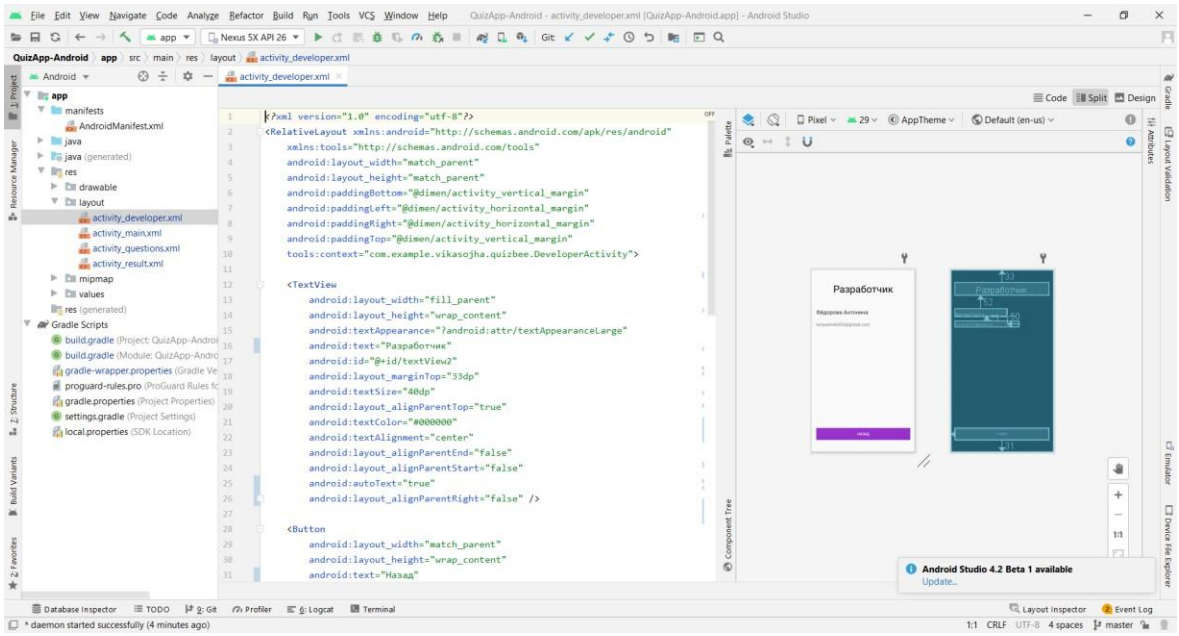

Figure 2.1 – Appearance of the Android Studio IDE – feature – on – feature dependencies (Figure 2.2).

– build (Gradle 4.0 plugin includes support for Android Studio's Build Analyzer (regardless of minimum API level) and creates dependencies between dynamic function modules;

– Java 8 and desaccharification in D8/R8 (previous versions of Android Gradle supported various Java 8 language features for all API levels through a process called desaccharification. Now you can use standard language APIs that were previously only available in recent versions of Android) [12].

When using the Gradle plugin 4.0.0 and above, you can specify that a Dynamic Feature module depends on another module. Being able to define this relationship ensures that your application has the necessary modules to unlock additional functionality, resulting in fewer requests and a simpler application structure.

For example, the function: video may depend on the function: camera. If the user wants to unlock the ability to record video, the application will automatically load the module: camera when requesting: video;

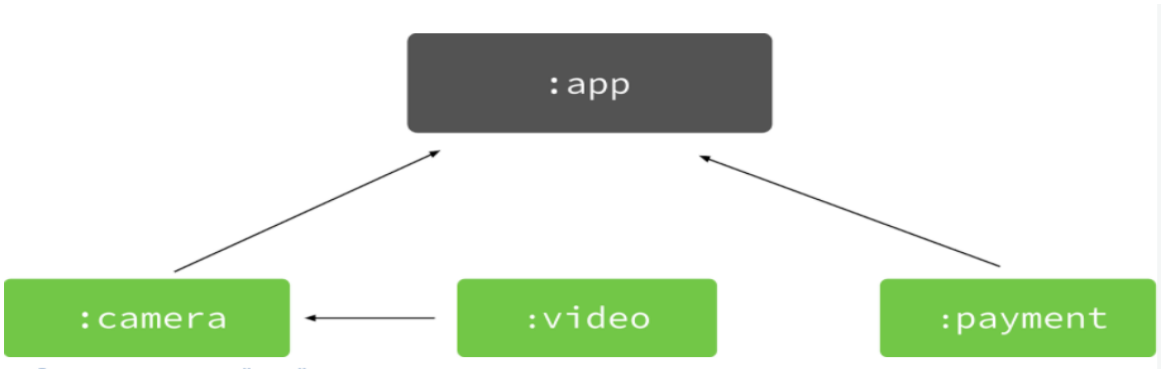

Figure 2.2 – Property dependency diagram

– support for Kotlin DSL files (the studio has built-in support Kotlin DSL build script files (*.kts). Kotlin build scripts offer a full set of quick fixes and are supported by the Project Structure dialog);

– dependency metadata (when creating an application, Gradle includes metadata describing the dependencies of compiled libraries).

When you upload an app, the Play Console checks this metadata, obtaining information about known issues with the SDK and dependencies used in the app, and in some cases providing feedback to resolve these issues.

Each project in Android Studio contains one or more modules with source code files and resource files. Types of modules include:

– modules – applications for Android;

– library modules;

– Mechanism modules – Google applications.

By default, Android Studio displays your project files in the Android Project view (Figure 2.3). This view is organized by modules to provide quick access to key project source files.

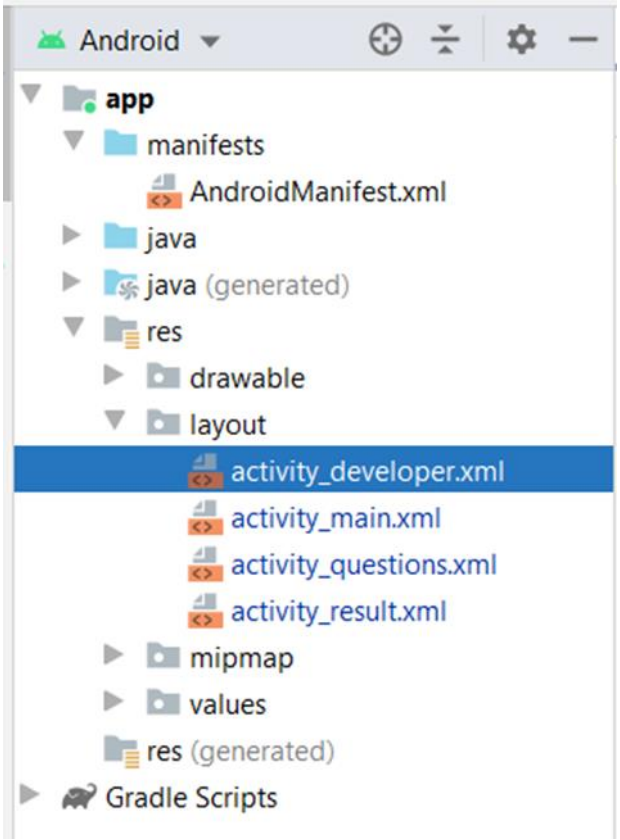

Figure 2.3 – Project view in Android Studio

All files of type "build" are visible at the top level according to Gradle scripts, and each application module contains the following folders:

– manifests: contains the AndroidManifest.xml file;

– java: contains Java source code files, including test code

JUnit;

– res: contains all non-code resources, such as XML markup, UI strings (user interface) and bitmap images.

Android Studio supports many version control systems (VCS), including Git, GitHub, CVS, Mercurial, Subversion, and Google Cloud Source Repositories.

2.2.2. Microsoft Visual Studio integrated environment

The Visual Studio integrated development environment is a launching pad for writing and debugging code, as well as further publishing applications.

An integrated development environment (IDE) is a multifunctional program that can be used for various aspects of software development. In addition to the standard editor and debugger that exist in most IDEs, Visual Studio includes compilers, code completion tools, graphical designers, and many other features to simplify the development process [13].

Some of the main tools of the environment:

– Solution Explorer lets you view, navigate, and manage code files. Solution Explorer lets you organize code by grouping files into solutions and projects;

– in the editor window (center), the contents of the file are displayed. Here you can edit the code or develop the user interface, such as a window with buttons or text fields;

– Team Explorer allows you to track work elements and share code with other users using version control technologies such as Git and the Team Foundation Version Control System (TFVC).

Visual Studio features:

– Visual Studio offers a set of tools that allow you to easily create cloud applications based on Microsoft Azure. It simplifies the setup, build, debug, package, and deploy of applications and services to Microsoft Azure directly from the IDE;

– Build web applications using ASP.NET, Node.js, Python, JavaScript, and TypeScript. Visual Studio recognizes web application frameworks such as Angular, jQuery, Express, and others. ASP.NET Core and .NET Core are supported on Windows, Linux, and Mac computers. ASP.NET Core is a major update to MVC, WebAPI, and SignalR that runs on Windows, Mac, and Linux. The ASP.NET Core platform was designed from the ground up and provides a compact and scalable .NET stack for developing modern cloud-based web applications and services;

– Using Visual Studio, you can create applications and games for macOS, Linux, and Windows platforms, as well as for Android, iOS, and other mobile devices.

– Server Explorer lets you view and manage SQL Server instances and resources in your local or remote environment, in Azure, Salesforce.com, Microsoft 365, and on websites.SQL Server Data Tools (SSDT) is a powerful development environment for SQL Server, SQL Azure Database, and SQL Azure Data Warehouse. With SSDT, you can create, maintain, debug, and refactor databases. You can work with a database project or directly with a connected database instance (locally or remotely).

2.2.3. Eclipse Java IDE

Eclipse is a free, open-source software platform controlled by the Eclipse Foundation. It is written in the Java programming language and its main purpose is to improve the productivity of the software development process. IDEs developed

on the basis of the Eclipse platform are used to create software in various programming languages, since Eclipse is a platform for developing any integrated programming environments and extensions for itself, according to the principle "Applications for Eclipse are developed in Eclipse itself."

Features of the Eclipse platform:

– cross-platform - works under Windows, Linux, Solaris and Mac OS X operating systems. Using Eclipse, you can program in many languages, such as Java, C and C++, PHP, Perl, Python, Cobol and others;

– is a framework for developing other tools and offers a large set of APIs for creating modules;

– Using the RCP (Rich Client Platform) approach, Eclipse is a tool for creating almost any client softwaresoftware;

– Work on the Eclipse project is being carried out in several directions, the main three being work on the Eclipse platform, development of a Java IDE, and development of plugins to extend Eclipse functionality;

– flexibility and extensibility is achieved thanks to the modularity of the platformrms.

Eclipse platform architecture:

– The main element is the Eclipse Runtime, in which the codes of extensions and modules are executed. It provides all the basic functionality of the platform for managing extensions and updates, interaction with the operating system, and ensuring the operation of the help system;

– the next element is the IDE itself – it is responsible for managing the main elements of the program, their location and settings, and managing projects, debugging and compiling projects, file searching, and team development.

– The standard Eclipse SDK includes two plugins – Java Development Tools or JDT, and Plug-in Developer Environment or PDE,

Thus, we get a fully-fledged IDE for Java programming and for developing extensions for Eclipse.

The Eclipse SDK is a minimal version, ideal for getting to know the platform and learning. Later, once you have decided on your goals, you can download and use any build that is suitable for your tasks, already equipped with the necessary extensions.

Eclipse also has a paid version, MyEclipse, but it is more of a standalone project built on top of Eclipse. MyEclipse offers a full-featured software development platform, as well as additional packages.

Eclipse, unlike the paid version, is a core to which additional plugins are connected to create the IDE of your dreams.

2.2.4. Comparative analysis of the presented IDEs

Android Studio is a new Android development environment based on IntelliJ IDEA. Similar to Eclipse with Android Development Tools for Eclipse, Android Studio provides integrated Android development tools for development and debugging.

The best IDE available for Android development with limited hardware capabilities on the development device.

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs for Microsoft Windows, as well as websites, web applications, and web services. Visual Studio uses Microsoft's software development platforms, such as the Windows API, Windows Forms, Windows Presentation Foundation, Windows Store, and Microsoft Silverlight. It can create both native code and managed code.

Visual Studio supports various programming languages and allows the code editor and debugger to support (to one degree or another) almost any language programming, if a language-specific service exists. Built-in languages include C, C++, and C++/CLI (via Visual C++), VB.NET (via Visual Basic .NET), C# (via Visual C#), and F# (from Visual Studio 2010). Support for other languages such as

M, Python, and Ruby, among others, is available through language services installed separately. It also supports XML/XSLT, HTML/XHTML, JavaScript, and CSS.

A comparative analysis of all three IDEs is shown in Table 2.1.

To create the Quiz game, the Android Studio IDE was chosen, and to create the crossword puzzle, Visual Studio was chosen.

Table 2.1 – Comparative analysis of IDEs

IDE	Android Studio	Microsoft Visual Studio	Eclipse
1	2	3	4
Platforms	Mac Windows Linux	Mac Windows .NET Framework Internet	Mac Windows Linux
Opportunities	Programming Code completion Integrated Android emulator	Auto fill Code completion Built-in Debugger API for development Color coding C# Scripting NuGet Exe Compiler Refactoring	Extensible with plugins/extensions Auto fill Java Development Environment Refactoring SCM Integration
Code	Open	Closed	Open
Price	Free	Community – free Enterprise – $250/month Professional – $45/month	Free

Eclipse is an extensible development platform with a runtime and application framework for building, deploying, and managing software throughout the software lifecycle. Many people know Eclipse as a Java IDE, but it is much

more than Eclipse. In fact, Eclipse consists of over 60 different open source projects organized into 6 different categories:

- entrepreneurship development;
- implementation and development of devices;
- rich client platform;
- rich internet applications;
- application frameworks;
- application lifecycle management (ALM);
- service-oriented architecture (SOA).

3. INFORMATION SYSTEM MODELING

3.1. Designing application information system using IDEF0 business process description diagram

IDEF0 is a methodology for graphically describing the systems and processes of an organization's activities as a set of interdependent functions. It allows you to study the functions of an organization without linking them to the objects that ensure their implementation. In the IDEF0 standard, input shows objects - information and material flows that are transformed in a business process. Control shows objects - material and information flows that are transformed in the process as needed for its execution. Using IDEF0 mechanisms, you can display the tools and resources with which the business process is implemented (for example, technical means, people, information systems, etc.). The output of the business process described in the IDEF0 standard fully corresponds to the content of the output of the process described using the DFD diagram [14].

The main goal and business process of the application is to create a quiz in single-user mode. The context diagram of this business process in the IDEF0 standard is shown in Fig. 3.1.

Context diagram description:

– business process name: single-user mode;
– main participants: user;
– input event: request to select a quiz subject; answers to quiz questions;
– output event: quiz result.

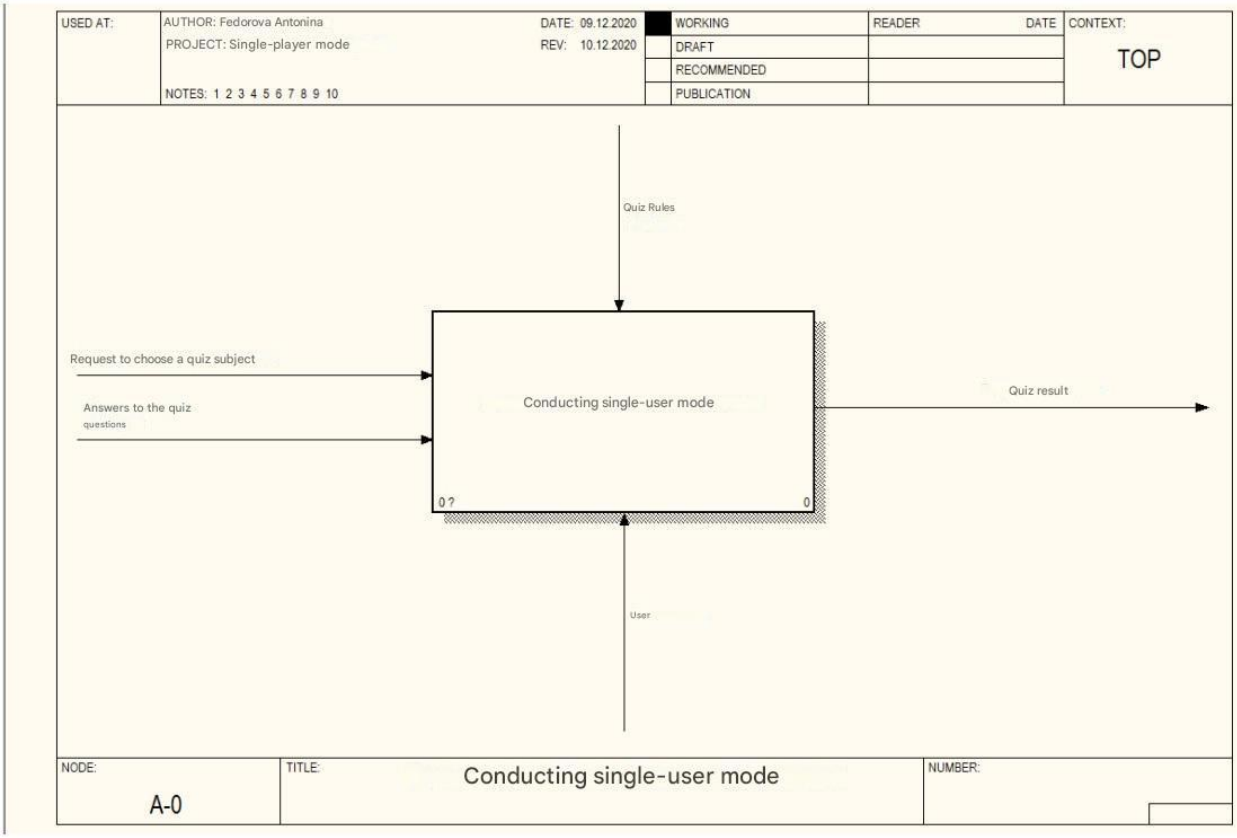

Figure 3.1 – Context diagram of the business process “Implementing single-user mode”

As a result of the decomposition of the context diagram, the following points were highlighted:

– bots:

– quiz formation;

– analysis of response results.

The diagram of the first level of decomposition is shown in Fig. 3.2.

Description of the first level of decomposition:

– Characteristics of the business process "Quiz Formation":

- main participants: user;

- input event: request to select a quiz subject;

- source event – selected quiz questions;

– Characteristics of the business process “Analysis of response results”:

- main participants: user;
- incoming event: selected quiz questions; answers to quiz questions;
- output event: quiz result.

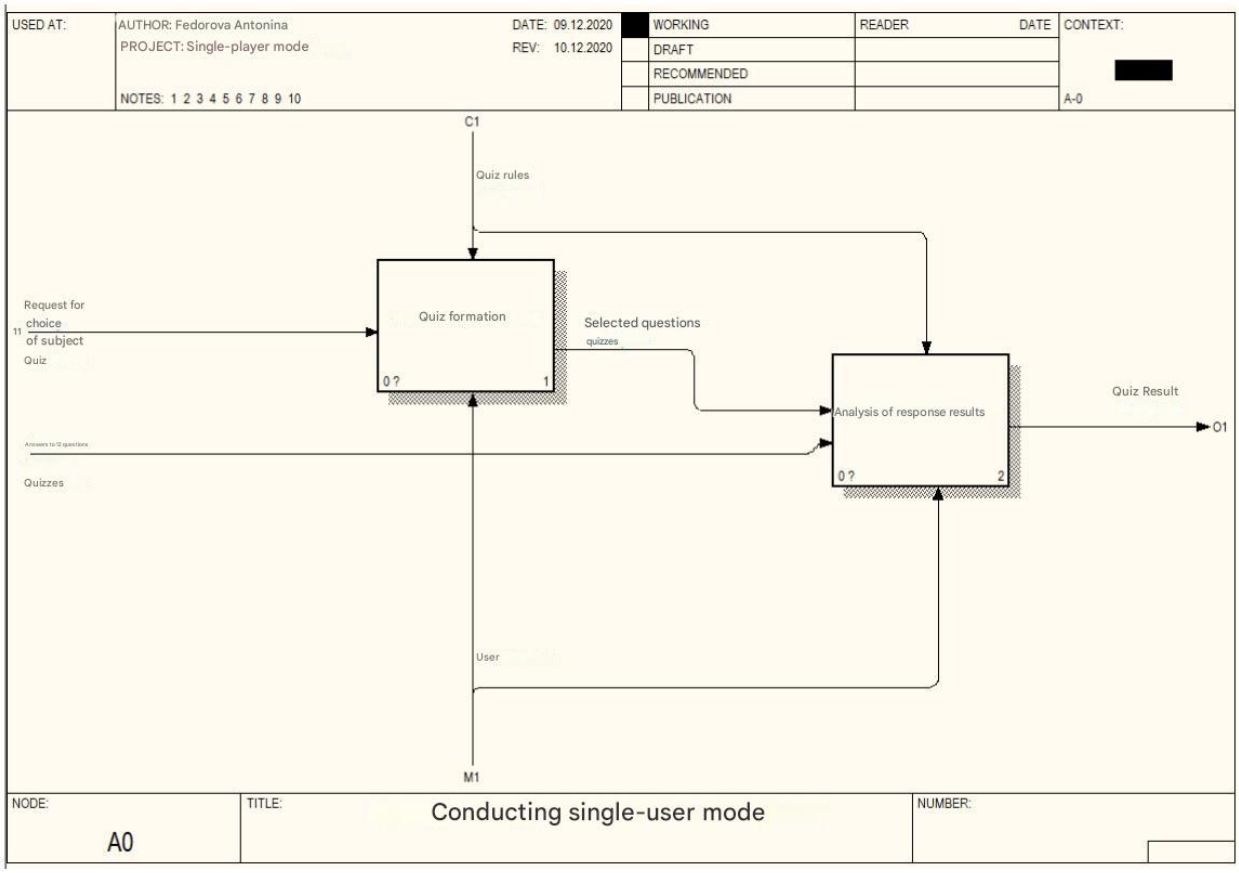

Figure 3.2 – The first level of decomposition of the business process “Implementing a single-user mode”

As a result of the decomposition of the business process "Quiz Formation", the following work was identified:

– choosing a quiz subject;

– getting quiz questions.

The decomposition diagram of the business process “Quiz Formation” is shown in Fig. 3.3.

Description of the decomposition of the business process "Quiz Formation":

– Characteristics of the business process "Quiz subject selection":

- main participants: user;
- input event: request to select a quiz question;
- source event: quiz subject;

– Characteristics of the business process "Receiving quiz questions":

- main participants: user;
- input event: quiz subject;
- source event: selected quiz questions.

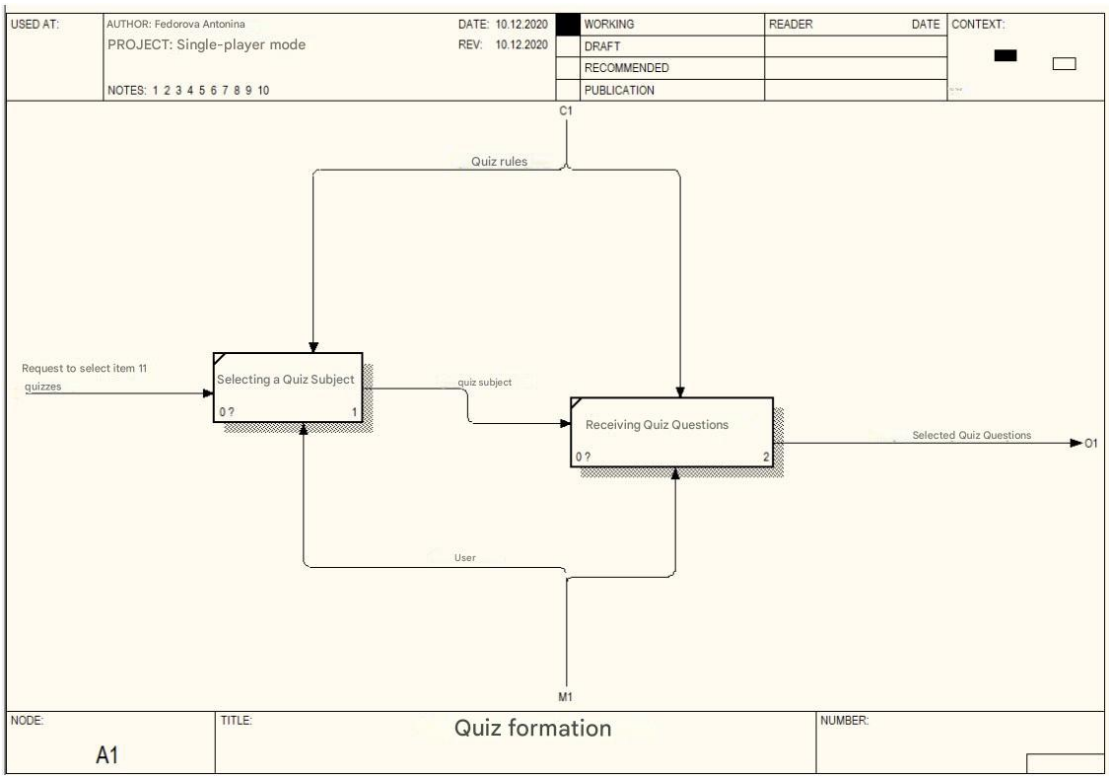

Figure 3.3 – Decomposition of the “Quiz Creation” business process

As a result of the decomposition of the business process “Obtaining quiz answers”, the following work was identified:

– providing answers to the quiz;

– response processing;

– formation of results.

The decomposition diagram of the business process “Analysis of quiz results” is shown in Fig. 3.4.

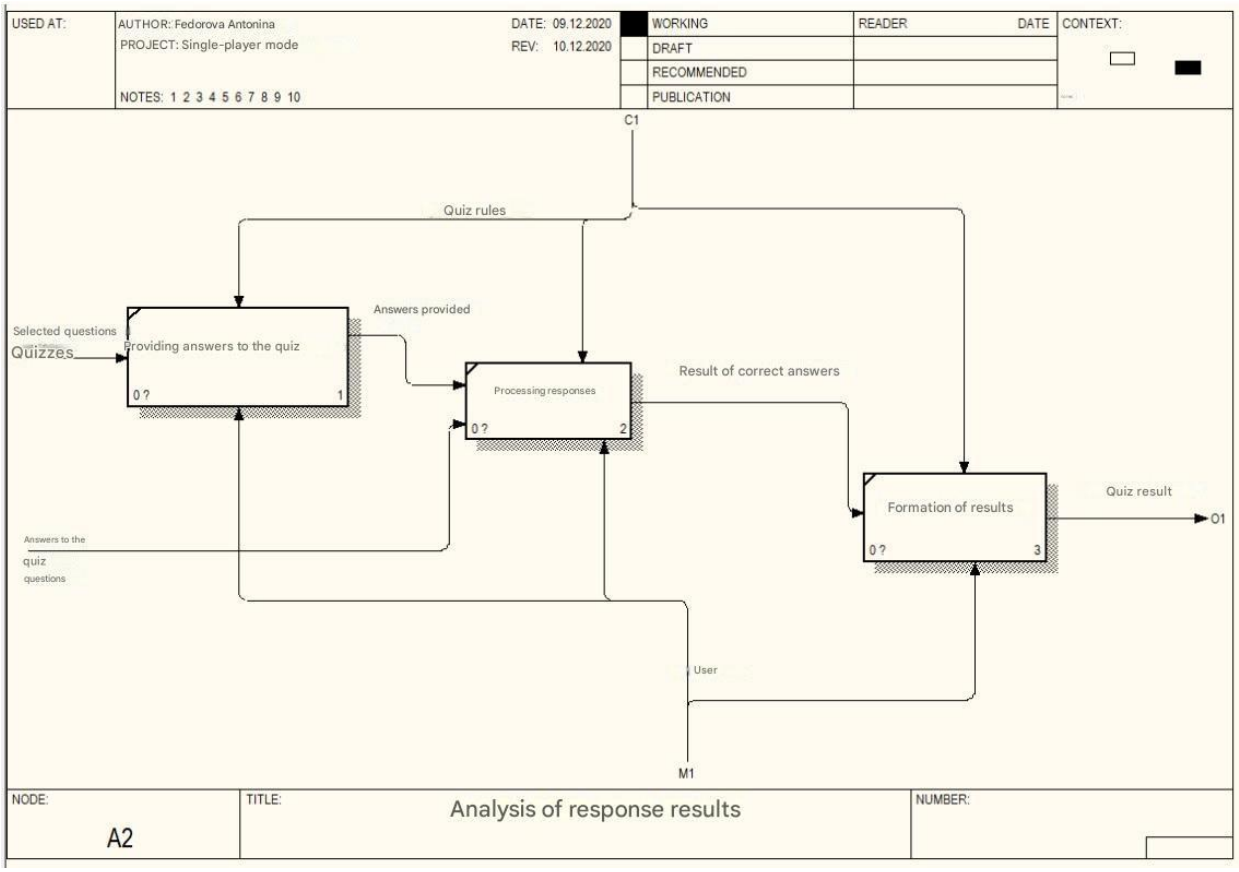

Figure 3.4 – Decomposition of the business process “Analysis of quiz results”

Description of the decomposition of the business process “Analysis of quiz results”:

– Characteristics of the business process "Providing answers to the quiz":

- main participants: user;
- input event: selected quiz questions;
- output event: answers provided;

– Characteristics of the business process "Response Processing":

- main participants: user;
- input event: answers provided; answers to quiz questions;

- output event: the result of correct answers;

– Characteristics of the business process "Results Formation":

- main participants: user;
- input event: result of correct answers;
- output event: quiz result.

The block diagram of the mobile application is presented in Fig. 3.5.

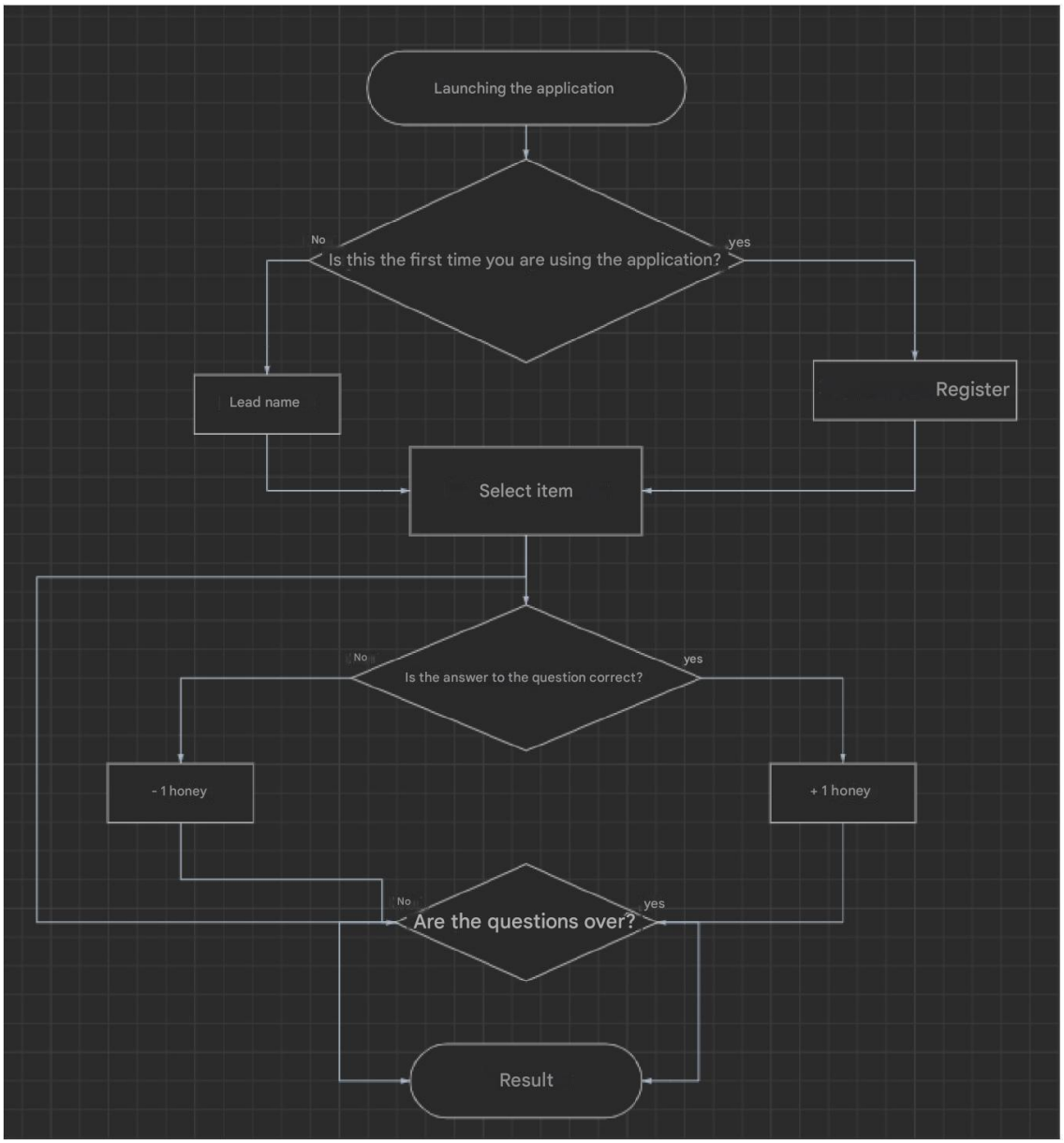

Figure 3.5 – Block diagram of the “Check Yourself” mobile application

3.2. Developing a use case diagram

To develop use cases, it is necessary to build a use case diagram for the application being developed (Figure 3.6). Use case diagrams describe the functional purpose of the system, or, in other words, what the system will do during its operation.

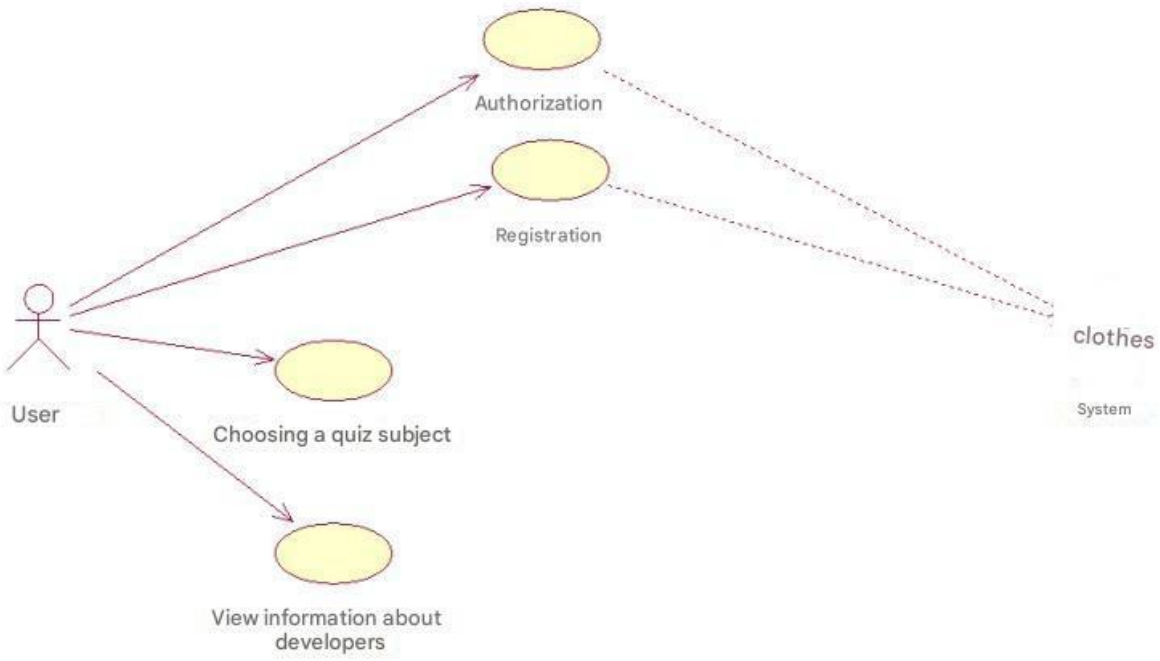

Figure 3.6 – Use case diagram

A use case diagram is an initial conceptual representation or conceptual model of a system during its development and implementation.

The essence of this diagram is as follows: the designed system is represented as a set of entities or actors that interact with the system using the so-called use cases. In this case, an actor or a person is any entity that interacts with the system from the outside. This can be a person, a technical device, a program or any other

system that can serve as a source of influence on the modeled system as determined by the developer himself.

In turn, a use case describes the services that the system provides to an actor. In other words, each use case defines a specific set of actions that the system performs when interacting with an actor. It does not say anything about how the actors will interact with the system.

During the design of the usage diagram, the following main use cases were identified:

– Quiz Subject Selection. The user selects the quiz subject.

– User authorization. The user authorizes the system so that when the result is saved, his name is in the record table.

Characteristics of the use case "Quiz subject selection":

– context of use: choosing a quiz subject;

– Actors: user;

– prerequisite: the user is authorized;

– trigger: the user wants to select a quiz subject;

– scenario: go to the main program window, select the subject:

– postcondition: if the use case is executed successfully, the quiz topic is selected, otherwise, there are no changes.

Characteristics of the "User Authorization" use case:

– usage context: user authorization;

– Actors: user;

– prerequisite: the user is registered;

– trigger: user wants to register;

– scenario: open the application, enter your username and password;

– postcondition: if the use case is executed successfully, the user enters the main menu, otherwise, a message appears stating that the login or password is incorrect.

3.3. Mobile application database design

The main component of a database is a table. A table is used to structure and store information. Within a database, there are relationships between tables, each of which specifies the sharing of table data. Di-

The Entity-Relationship diagram graphically represents the data structure of the designed system [15].

Having determined the functional requirements, using SQL modeling, a database was designed that implements the Entity-Relationship model. During the development of the system, a database was created using the Microsoft SQL Server DBMS.

The first stage of the database design process is called conceptual database design. It consists in creating a conceptual data model of the subject area. This data model is created based on the functional requirements of users. Conceptual database design is completely independent of such details of its implementation as the type of selected DBMS, the set of created application programs, the programming languages used, the type of selected computing platform, as well as any other features of the physical implementation. Conceptual design is the creation of a conceptual representation of the database, which includes the definition of the types of the most important entities and the relationships and attributes existing between them. The sequence of stages of designing a conceptual data model: definition of entities; definition of relationships between entities; definition of attributes of entities; assignment of primary and alternative keys.

3.3.1. Description of database tables and relationships between them

For the information system, the following entities were defined in the database: User, Subject, Question, Result. Each entity contains a primary key

designed to identify an instance of the entity. The primary key must be chosen in such a way that the values of the attributes can accurately identify the entity, and none of the attributes of the primary key must be primary key must not have a null value. The values of the primary key attributes must not change.

To define the primary key, unique numbers are often used, which can be automatically generated by the system when an entity instance is added to the database. The use of unique numbers facilitates the process of indexing and searching in the database.

Next, you need to define the attributes for each entity, and also write down the primary keys for all entities. Now, set up the relationships between the entities [16]. Let's imagine the mobile application database in the form of an Entity-Relationship model (Fig. 3.7).

The User entity contains the following attributes: number (primary key), username, login, password.

The Subject entity contains the following attributes: number (primary key), subject name.

The Question entity contains the following attributes: number (primary key), question, answer 1, answer 2, answer 3, correct answer, incorrect answer.

The Result entity contains the following attributes: number (primary key), number of correct answers, number of incorrect answers, result.

There is a one-to-many relationship between the User and Subject tables: one user can have multiple subjects.

Let's consider the data types of each entity attribute.

Information about data types is given below in the form of tables (Table 3.1-3.5).

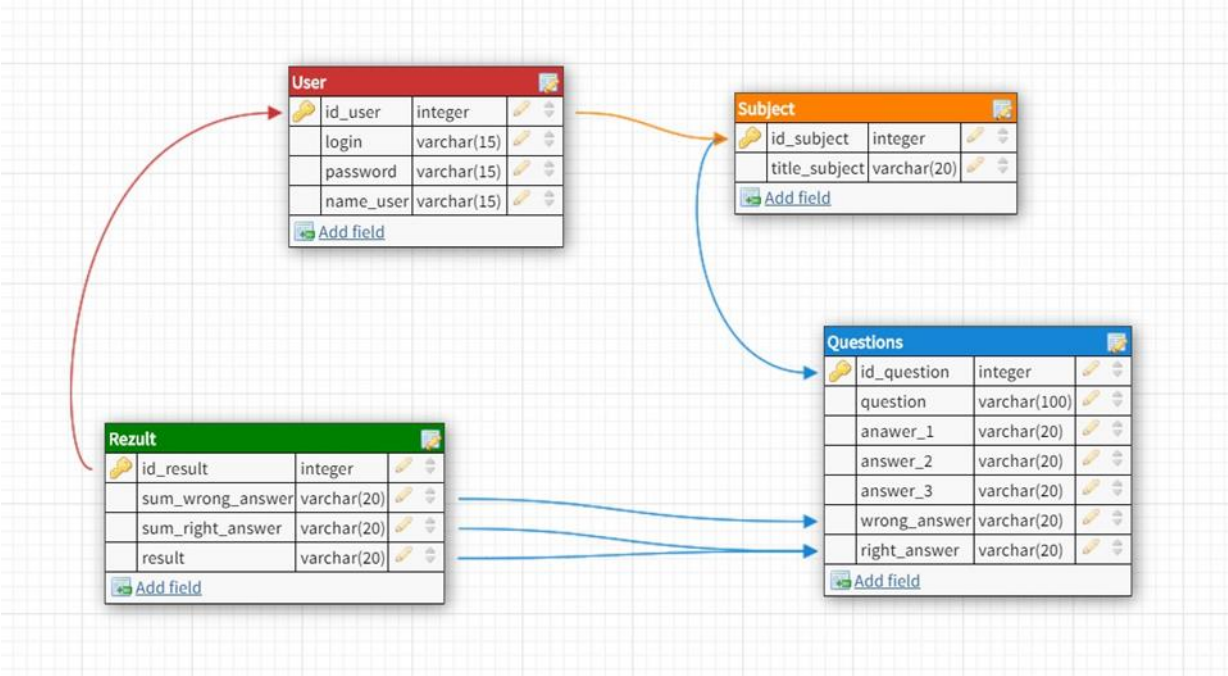

Figure 3.7 – Entity-Relationship Diagram of the "Check Yourself" Mobile Application

Table 3.1 – List of all entities in the information system database

User	User
Subject	Subject
Questions	Question
Result	Result

Table 3.2 – User (user)

No.	Field	Attribute	Type
1	user_id	Unique user identifier	integer
2	username	User name	varchar(15)
3	login	User login	varchar(15)
4	password	User password	varchar(15)

Table 3.3 – Subject

No.	Field	Attribute	Type
1	id_subject	Unique item identifier	integer
2	title_subject	Subject name	varchar(20)
3	username	User name	varchar(15)

Table 3.4 – Questions

No.	Field	Attribute	Type
1	id_question	Unique question identifier	integer
2	question	Question	varchar(100)
3	answer_1	Answer 1	varchar(20)
4	answer_2	Answer 2	varchar(20)
5	answer_3	Answer 3	varchar(20)
6	wrong_answer	Wrong answer	varchar(20)
7	right_answer	Correct answer	varchar(20)

Table 3.5 – Result

No.	Field	Attribute	Type
1	id_result	Unique result identifier and	integer
2	sum_wrong_answer	Number of incorrect answers	varchar(20)
3	sum_right_answer	Number of correct answers	varchar(20)
4	result	Final result	varchar(20)

3.4. Crossword generation algorithm

3.4.1. Exhaustive search method

Exhaustive search algorithms are used when there are no others ways to solve a certain problem. For example, no one would think of solving a quadratic equation by searching for all possible values, even if it is known that the solution is integers. Exhaustive search is a "head-on" solution, built on the substitution and verification of all possible options from the domain of permissible values. In words, there is nothing complicated in solving an exhaustive search problem, but in reality, algorithms that really search and find solutions in this way in a "reasonable time" are quite difficult. The most popular problems, the solution of which is considered in schools and universities by the exhaustive search method (within the framework of disciplines related to combinatorics and programming), include filling a rectangular region with Tetris figures, finding a way out of a maze, the backpack problem, finding related regions, and others. Many graph problems can also be solved using exhaustive search, although special methods have been invented for their solution - algorithms that are optimal in terms of the number of operations and search for solutions an order of magnitude faster than the usual exhaustive search. Such problems include finding the shortest route between vertices (Dijkstra's algorithm) or finding the maximum throughput in a graph. To summarize all of the above, to solve a problem using exhaustive search, it is necessary to develop an algorithm for searching all possible solution candidates and implement a procedure for checking the correctness of the solution for each of them.

The computational complexity of exhaustive search algorithms is estimated by the following generalized formula:

$$O\ (N!\ /\ (NR)!) \qquad (3.1)$$

where, N is the number of elements in the solution search area.

R is the number of elements that form a single solution to the problem. If we return to the problem of finding all options for filling in the cross-word, then its complexity, in the simplest version, can be estimated as follows:

$$N * (N-1) * ... * (N-(W-1)) \quad (3.2)$$

where, N is the number of words in the dictionary, and W is the number of word patterns in the crossword. At each subsequent step, one word less than the previous one should be selected, since it is impossible to repeat the selection of words. And this is nothing more than:

$$N! / (NW)! \quad (3.3)$$

Naturally, there should be no fewer words in the dictionary than there are their templates in the crossword, otherwise there are no solutions at all. If there are more than ten words in the crossword, and in the dictionary for filling it, for example, more than a thousand words, then the total number of options that need to be sorted out to find all solutions will amaze even "experienced" supercomputers with the number of digits. From all this, one thing follows: if the problem cannot be solved in any way except by exhaustive search, then the algorithm for this exhaustive search must be "optimal". I put the word "optimal" in quotes, since rigorously proving the optimality of such algorithms is even more difficult than developing the algorithms themselves. In the context of this article, instead of the term "optimal", I will use the term "satisfactory". We will consider an algorithm satisfactory that finds at least one solution to the problem in a time that is satisfactory for the user. This term is "fuzzy" and has little in common with the exact sciences to which algorithmization belongs, but in our case we don't have to choose.

3.4.2 Optimization of the exhaustive search algorithm

Optimization of any algorithm should be carried out in at least two directions: in the direction of minimizing the number of elementary operations and in the direction of optimizing the speed of execution of these elementary operations. The second direction is mainly related to the optimization of the data structures used and the operations to access them. The most popular methods of reducing the execution time of elementary operations are caching of frequently used data, postponing initialization and various kinds of calculations until the moment when they are needed, bitwise comparison, etc. It is necessary to understand that memory consumption and the speed of execution of operations are two opposite characteristics. By creating additional "tricky" data structures in RAM, you can increase the speed and vice versa: if you do not save the results of intermediate calculations and try to do all the necessary calculations anew each time, you can win in memory volume, but significantly lose in the speed of the program. I admit, I didn't do any special research related to this kind of optimization, I just tried not to make unnecessary movements and actions.

Exhaustive search algorithms are often implemented using recursion: at each step of the recursion, a part of the solution is searched for for one element, and once it is found, the same search is recursively launched for the next element. If a solution cannot be found for the current element, then a transition is made to the previous element (returning from recursion one step), where the found solution is canceled and the search for another option begins.

Once solutions are found for all elements, all solutions are found. Then, the algorithm saves the solution variant and continues to search for others. During the analysis of a recursive exhaustive search algorithm, it is very important to be able to assess whether the current recursion branch contains a solution or can be cut off.

For example, is it worth trying to arrange other words in a crossword puzzle when you come across an unsolvable intersection: there are no words in the dictionary of the required length for substitution into the template that meet the intersection conditions? Analysis and cutting off deliberately "unproductive" branches of recursion is called the branch and bounds method. Optimization of exhaustive search algorithms can be carried out using a heuristic approach to solving the problem. This approach is associated with a set of assumptions, the correctness of which is either difficult or formally impossible to prove, but adhering to them allows you to find a solution quite quickly. Let me remind you that the goal of the optimization being undertaken is to find at least one solution as quickly as possible, and not all options for filling in the crossword puzzle. Thus, optimization of the recursive exhaustive search algorithm, aimed at minimizing the number of elementary operations (comparison, selection, etc. operations), is carried out by applying two approaches:

– searching and cutting off recursion branches that deliberately do not contain solutions – the branch and bounds method;

– The choice, first of all, of those recursion branches that contain the largest number of solutions is a heuristic approach.

3.4.3. Optimization of the crossword generation algorithm

In a nutshell, the presented algorithm does the following: first, tav-generates words in one direction – the first line (GenerateFirstLine), and then tries to arrange words in another direction taking into account intersections – the second line (GenerateSecondLine). It is clear that when filling the first line, no intersections need to be taken into account, but it is enough to simply select words of the length corresponding to the template.

The first optimization is optimization of the data structures used: divide the entire dictionary into sections, each of which contains words of only one type

length, and organize quick access to the section by a given length. This is done using a dictionary based on a hash table, the values of which are lists of words - sections, and the key is the length of words in the section. Since the selection of words in the first line is performed faster (intersections are not taken into account), the first line is the direction with the largest number of word patterns.

The presented algorithm is recursive: at each step, a search is made for a suitable variant for a specific word template and, if the search is successful, a transition is made to the next template. The solutions for the sequence of steps within one line are weakly dependent, since they are connected only by checking whether a specific word from the dictionary was used when filling the template in the previous steps (filling several word templates with the same value is prohibited!). But the steps of the second line are very, very dependent on the solutions found in the first line.

The second optimization is a heuristic approach. We assume that if the word templates filled in the first line are sorted in ascending order, and the word templates in the second line in descending order, this will help to find the first ready-made version of the filled crossword faster. The motivation is as follows:

The largest number of returns to previous recursion steps will occur at the boundary of the first and second lines.

The longer the length of the current word pattern, the more seriously its value affects the search progress for all remaining solutions.

Thus, the longest word patterns are concentrated near the border of two lines of the recursive algorithm. Word searching when trying to fill long patterns will lead to a rapid change in the solution search areas, and therefore will increase the probability of encountering at least one of them at the very beginning of the full search of options.

The third optimization is the branch and bounds method. If the search for a value for the word pattern within the second line does not yield a result, on the previous line step, the template of the same line (same orientation) was filled, then

there is no point in returning to the previous recursion step if the length of the previous template is greater (remember that the lengths of the templates are sorted in descending order). There is no point in this, since in this case all the previous solutions of the second turn are independent of the solution for the current word template. It makes sense to return to the template of the first turn with which there is an intersection. If there are several intersections, then we choose the nearest recursion step - the template with the largest ordinate.

3.5. Software realization informational systemsinteractive learning methods

3.5.1. Implementation of the “Check Yourself” mobile application

When the program is launched (Fig. 3.8), the main activity is launched, and the user is given the choice of registering or entering a name (for registered users).

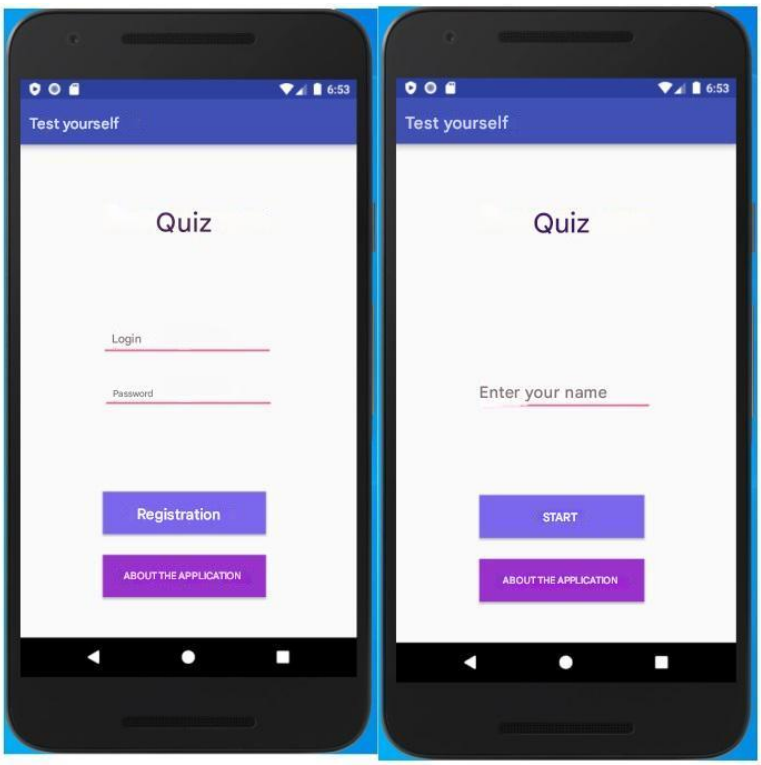

Figure 3.8 – Launching the application

If the application is launched for the first time, the User is prompted to register (enter username, password and click the "Register" button).

If the application has already been launched, just enter your name and click the "Start" button.

There is also an “About the application” button on the main page. If you click on it, a window will appear with information about the developer (Fig. 3.9).

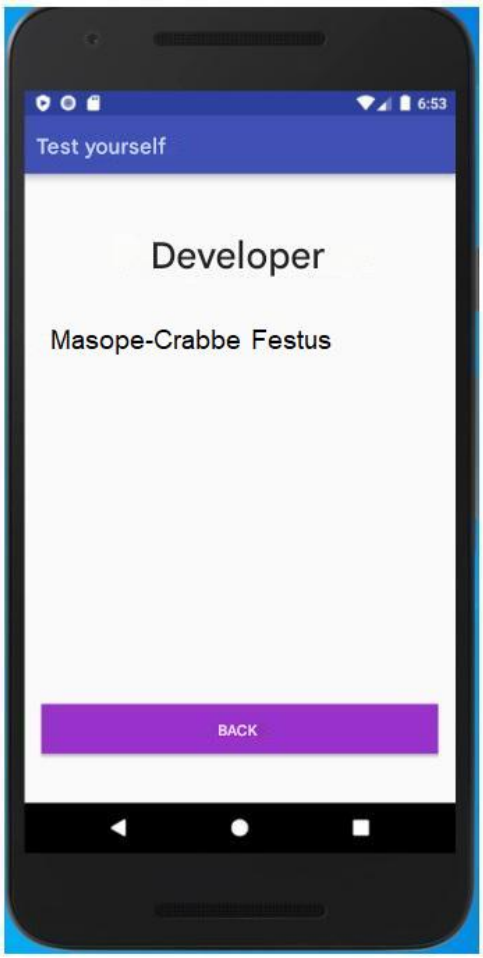

Figure 3.9 – “About the application” window

After the User has entered their data, they need to click the button

"Start". After that, the subject selection window opens, testing with which the User will pass (Fig. 3.10). Subjects to choose from: programming, neural networks, OOP, Web design, modeling.

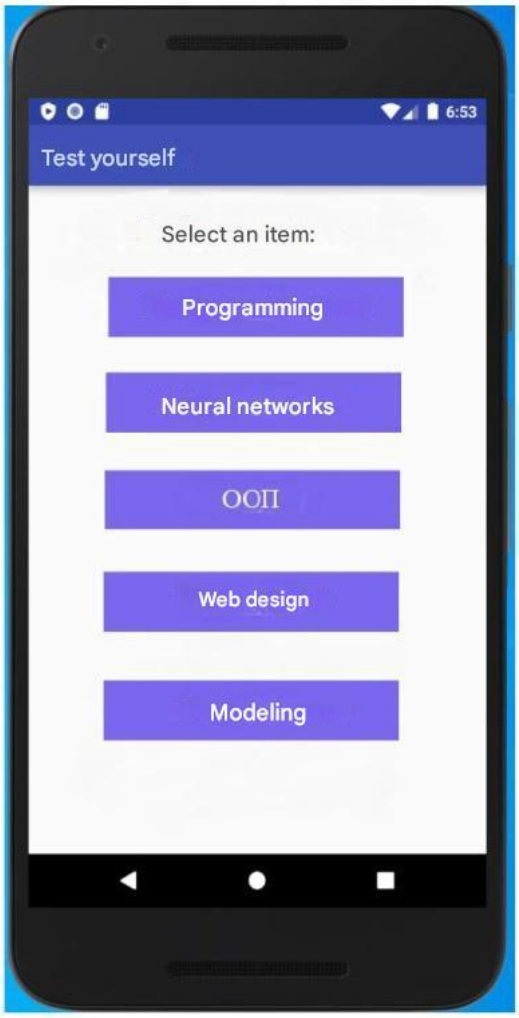

Figure 3.10 – Subject selection window “Select subject”

After the User has selected the subject to be tested, testing begins (Fig. 3.11).

The testing window contains the following elements:

– greeting the User;

– question;

– 3 answer options;

– the "Next Question" button, which directs the user to the next testing page;

– "Exit" button to exit the application;

– and below – the number of points received at the time of testing.

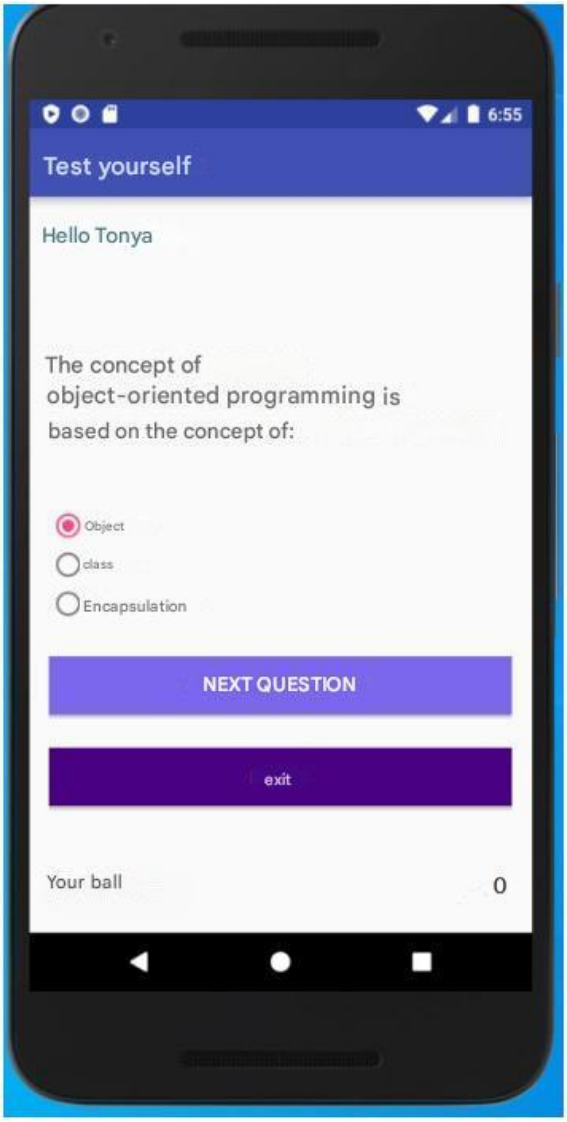

Figure 3.11 – Testing window.

After the User has passed the test, a results window appears (Fig. 3.12).

The results window shows:

– number of correct answers;

– number of incorrect answers;

– final score;

– Restart button.

Figure 3.12 – Results window.

3.5.2. Implementation of the Crossword project

The classes and structures representing the crossword model are grouped together in the Model namespace. Other elements are located either in the root project namespace or in the Properties namespace.

Grid class. Crossword template. This class represents nothing more than a crossword template: a rectangular array of logical. The class's tasks include very few things: changing the template size, editing cells (inserting a cell for a symbol or an empty space), and generating a list of word templates, both horizontally and vertically.

The buildWordList method. In it, the StreamReader class constructor implements reading text from a file. The String.Split method creates an array of substrings by splitting the input string by one or more delimiters.

InitializeBoard method. It forms the crossword grid. Method

ToUpper converts text to uppercase. The formatCell method formats whitespace.

cells. The board_CellPainting method draws the grid and cells and fills them with colors.

The Form form is the main form and initializes and displays the crossword grid. The Clues form is a form of questions and clues for solving the crossword.

The Puzzles folder stores a file that contains all the information about the crossword: how to fill it in: horizontally or vertically, the question itself, and the answer.

Application infrastructure. This group includes additional application components that perform service roles. The Properties namespace contains the Resources and Settings components. The root namespace defines the Program component, which implements the application entry point, the static Main function.

Description of the program

After starting the program, 2 main windows appear on the screen (Fig. 3.13, Fig. 3.14).

The first window presented (Fig. 4.6) is a crossword grid to fill in. Each word, depending on the position, is numbered. If the word is correct, the letters are lit green, otherwise, they are lit red.

File tab – allows you to open any crossword file that has been prepared in advance, and also has an exit function.

The Help tab stores information about the developer and the product itself.

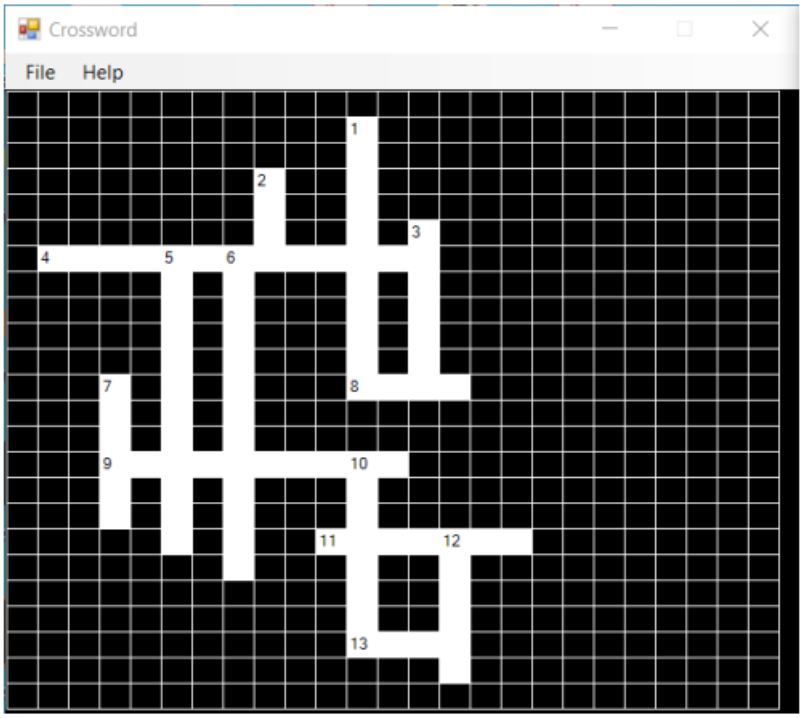

Figure 3.13 – Main program window

#	Direction	Clue
4	across	Responsible for programs
8	across	Responsible for managing data in internal and external memory and logging
9	across	The minimum logically meaningful operation that makes sense and can only be performed in full
11	across	A method of conveying content that involves organizing a data structure in which individual elements are placed into cells
13	across	Element responsible for text recognition software
1	down	The login procedure that a user must complete to gain access to a computer system
2	down	A computer that provides direct support for the server's operation.
3	down	A special type of stored procedure that is not called by the user, but whose use is conditioned by the occurrence of a certain condition.
5	down	Data Redundancy Reduction Process
6	down	One of the server tasks
7	down	Record in a relational database
10	down	Strict subordination of elements, one main, the rest subordinate
12	down	A database object created to improve the performance of data retrieval.

Figure 3.14 – Questions and Hints Window

The second window presented (Fig. 3.14) depicts the crossword question, clues to the location of the word (horizontally or vertically).

4 SAFETY OF LIFE, BASIC LABOR PROTECTION

4.1. Labor protection requirements when working with electrical equipment

General provisions

The labor protection instructions for an electrician when performing repair and maintenance work on electrical equipment were developed in accordance with the Law of Ukraine “On Labor Protection” (Resolution of the Verkhovna Rada of Ukraine dated 10/14/1992 No. 2694-XII) as amended on 01/20/2018, based on the “Regulations on the Development of Labor Protection Instructions”, approved by the Order of the Labor Protection Supervision Committee of the Ministry of Labor and Social Policy of Ukraine dated January 29, 1998 No. 9 as amended on September 1, 2017, taking into account the “Rules for the Technical Operation of Consumer Electrical Installations”, approved by the Order of the Ministry of Fuel and Energy dated July 25, 2006. No. 258 (as amended by the order of the Ministry of Energy and Coal Industry of Ukraine dated 13.02.2012 No. 91, “Rules for the safe operation of electrical installations of consumers”, approved by the order of the State Supervision Service of Ukraine dated 09.01.1998 No. 4.

All provisions of this labor protection instruction apply to electricians of an educational institution who perform repair and maintenance work on electrical equipment.

Persons not younger than 18 years old who have undergone training in the specialty and who are also allowed to perform repair and maintenance work on electrical equipment independently are:

a medical examination and do not have contraindications due to health to perform this work;

introductory and primary workplace briefings on labor protection;

training in safe methods and techniques of work;

testing of knowledge of the rules for installing electrical installations, safety rules for operating electrical installations, labor protection requirements;

when repairing and maintaining electrical equipment voltage up to 1000V have an electrical safety group not lower than III, and over 1000V - not lower than IV.

Electricians must know and comply with the requirements of the labor protection instructions when performing work on the repair and maintenance of electrical equipment, instructions for working with hand tools, power tools and ladders.

Electricians when performing work on the repair and maintenance of electrical equipment must comply with the requirements of the Rules for the safe operation of electrical installations of consumers and the Rules for the technical operation of electrical installations of consumers, and have an appropriate electrical safety group in accordance with the requirements of these Rules.

When performing work on the repair and maintenance of electrical equipment, the impact of the following harmful and dangerous production factors may be observed:

fall from a height;

electric shock;

increased electric field strength;

increased dustiness of the air in the work area;

increased vibration level;

insufficient illumination of the work area;

physical overload;

neuropsychic overload.

Electricians when performing repairs and maintenance of electrical equipment must use the following PPE:

cotton overalls - for 12 months;

gloves for - 3 months;

leather boots for - 24 months;

dielectric galoshes - on duty;

dielectric gloves - on duty;

dielectric mats - on duty.

An electrician when repairing and maintaining electrical equipment is obliged to:

keep his workplace clean and tidy;

comply with the Rules of Internal Labor Regulations;

be able to use personal and collective protective equipment, fire extinguishing equipment;

be able to provide first aid to accident victims;

know and comply with all requirements of regulatory acts on labor protection, fire protection rules and industrial sanitation.

immediately inform your immediate supervisor about any accident that occurred at work, about signs of an occupational disease, as well as about a situation that poses a threat to the life and health of people;

know the testing dates of protective equipment and devices, the rules for their operation, care and use. It is not allowed to use protective equipment and devices with an expired inspection period;

perform only the assigned work;

comply with the requirements of the equipment operating instructions;

know where the first aid facilities, primary fire extinguishing equipment, main and emergency exits, evacuation routes in the event of an accident or fire are located;

know the telephone numbers of a medical institution (103) and fire department (101).

An electrician may refuse to perform the work assigned to him if a production situation arises that poses a threat to his life and health of others, or to the environment, and report this to his immediate supervisor.

Smoking, drinking alcoholic beverages and other substances that have a narcotic effect on the human body are prohibited in the workplace.

In order to prevent injuries and the occurrence of dangerous situations, the following requirements must be observed: it is impossible to involve third parties in the work;

do not start work if there are no conditions for its safe performance;

perform work only on serviceable equipment, with serviceable devices and tools;

if a malfunction is detected, immediately report it directly to

to the manager or eliminate them on their own, if this applies to their job duties;

not to touch uninsulated or damaged wires;

not to perform work that is not part of their professional duties.

Be able to provide first aid for bleeding, fractures, burns, electric shock, sudden illness or poisoning.

Follow the rules of personal hygiene:

outerwear, hats and other personal belongings should be left in the wardrobe;

work in clean overalls;

eat in the designated place.

Be able to correctly use PPE and collective protection equipment, primary fire extinguishing equipment, fire-fighting equipment, know where they are.

Persons who violate this labor protection instruction for an electrician when performing repair and maintenance work on electrical equipment shall bear disciplinary, administrative, material and criminal liability in accordance with the current legislation of Ukraine.

Safety requirements before starting work

Wear overalls, inspect and prepare the workplace, remove unnecessary objects.

Remove unauthorized persons from the work area and clear the workplace of foreign materials and other objects, fence off the work area and install safety signs.

Make sure that the workplace is sufficiently illuminated, that there is no electrical voltage on the repaired equipment.

Inspect the serviceability of switches, electrical outlets, power cords, electrical wires, connecting cables, make sure that PPE (personal protective equipment) and warning devices (dielectric gloves, safety glasses, galoshes, mats, etc.) are available and in good condition.

When working with a tool, it is necessary to make sure that it is in good condition, that there is no mechanical damage to the insulating coating and that the tool has been tested in a timely manner.

Inspect the workplace for compliance with fire safety requirements and for adequate workplace lighting.

If you find any deficiencies or violations in electrical and fire safety, immediately report them to your immediate supervisor.

4.2. Safety requirements during work

When performing your duties, an electrician must have a certificate of knowledge testing on labor protection. In the absence of a certificate or a certificate with an expiration date, the employee is not allowed to work.

Work in electrical installations is divided into 3 categories in terms of safety measures:

with voltage relief;

without voltage relief on or near live parts;

without voltage relief away from live parts.

Employees performing special types of work that require additional safety requirements must be trained in the safe conduct of such work and have a corresponding entry in the knowledge test certificate.

An employee who serves electrical installations assigned to him with a voltage of up to 1000 V alone must have a III group on electrical safety.

When performing work in electrical installations, it is necessary to carry out organizational measures that ensure the safety of work:

draw up work orders-permits, orders in accordance with the list of works performed in the order of current operation;

prepare workplaces;

admittance to work;

exercise control over the performance of work;

transfer to another workplace;

establish breaks in work and its completion.

To prepare the workplace for work that requires voltage relief, it is necessary to apply, in a certain order, the following technical measures:

perform the necessary shutdowns and take all measures that exclude erroneous or unauthorized switching on of switching equipment;

hang prohibition posters on the drives of manual and remote control keys of switching equipment;

check for the absence of voltage on conductive parts that must be grounded to protect people from electric shock;

install grounding (turn on grounding knives, use portable grounding);

install fences, if necessary, near workplaces or live parts that remain under voltage, and also hang safety posters on these fences.

depending on local conditions, fence live parts before or after their grounding.

At least two workers should work without removing voltage on or near live parts, one of whom, the work supervisor, must have group IV; the others must have group III with mandatory registration of the work with a work permit or order.

When removing and installing fuses under voltage in electrical installations with voltage up to 1000 V, all loads connected to the specified fuses should be disconnected in advance; use insulating pliers or dielectric gloves, and if there are open fuse inserts, then safety glasses.

Work using ladders must be carried out by two people, one of the workers must be at the bottom. Standing on boxes or other objects is prohibited. P

When installing extension ladders on beams, elements of metal structures, etc., the upper and lower parts of the ladder should be securely fixed to the structures.

During maintenance and repair of electrical installations, it is prohibited to use metal ladders.

4.3. Safety requirements after completion of repair and maintenance of electrical equipment

Disconnect (disconnect) the necessary electrical equipment, power tools from the network.

Clean up the workplace, remove parts, material, garbage and waste to special places.

Remove all tools and devices to the designated place.

Remove and remove overalls, PPE, wash hands thoroughly.

Inspect the workplace for compliance with all fire protection requirements.

Notify your immediate supervisor of any deficiencies and malfunctions that occurred during the work. Record this in the operational log.

Safety requirements in emergency situations

In case of fire:

turn off electrical equipment, supply and exhaust ventilation, if any;

notify the fire department by calling 101 and report this to your supervisor, and in his absence, to another official;

proceed to eliminate the source of the fire, using the fire extinguishing agents provided for this purpose. Extinguish electrical equipment that is under voltage can only be extinguished with carbon dioxide fire extinguishers of the OU type or sand. It is prohibited to extinguish them with water or foam fire extinguishers.

The electrician must remember that in the event of a sudden power outage, it can be supplied again without warning.

Mechanisms and devices should be quickly turned off:

in the event of a sudden power outage;

if their further operation threatens the safety of employees;

in the event of a feeling of electric current when touching metal parts of the starting equipment;

in case of sparking;

at the slightest sign of ignition, smoke, or a burning smell;

if an unfamiliar noise appears.

In the event of a short circuit in the power supply network, it is necessary to de-energize the equipment and notify your immediate supervisor.

If an electric shock occurs, the victim should be released from the action of the electric current, for which purpose the electrical network should be turned off or the victim should be disconnected from the conductive parts using dielectric protective equipment and other insulating items and objects (dry clothing, dry pole, rubberized material, etc.), or the wire should be cut (chopped) with any tool with an insulating handle, carefully, without causing additional injuries to the victim. Before the arrival of a medical worker, it is necessary to provide the victim with first aid.

In the event of accidents (injury to a person), immediately notify the immediate supervisor.

CONCLUSIONS

The result of the first part of the thesis is a mobile game application for conducting quick and convenient interactive testing in the subject area. The developed application satisfies all the requirements that were identified during the analysis of the subject area and existing solutions.

While working on the diploma project, the following tasks were solved:

– a meaningful description and analysis of the subject area, structural and functional features of the management object was made: the activities of the management object were described, the business processes of the subject area were modeled in the IDEF0 standard;

– A review and analysis of existing systems was conducted.

– functional requirements were developed: use case diagrams were constructed;

– designed database structure.

The functionality of this application is sufficient to influence the quality of the educational process, namely:

– increase student engagement;

– improve the memorability of new material;

– reduce the workload on the teacher.

The result of the second part is a implemented software product – a crossword puzzle.

A crossword is a puzzle that consists of interweaving rows of cells that are filled with words with a given meaning.

Usually, the meanings of the words are given descriptively under this figure, first the meanings of the words that should appear horizontally, then vertically.

During the course of the thesis, a fully functional software for generating crosswords was obtained, which runs on the Windows operating system. The

interface of the created program is convenient, simple, and intuitive improves its capabilities

The developed application is a tool for interactively creating crossword forms and filling them in. In other words, it is a crossword generator. Similar applications can be used by various publications that either specialize in publishing crosswords or sometimes decorate their products with them.

The developed software satisfies all the requirements set at the task formulation stage.

REFERNCES

1. Arduino Robot. URL: http://www.arduino.cc/en/Main/Robot.
2. DSpace JSPUI (screenshot). URL:http://repo.knmu.edu.ua/hadle/123456789/8425. (date of application – 01.12.2024).
3. Overview of interactive methods (title from the screen). URL: llt.multycourse. com.ua/ua/page/19/67#2. (date of access – 01.12.2024).
4. Interactive methods teaching (title) with screen). URL: https://sites.google.com/site/nmcmyk/naukova-dialnist/interaktivni- method-navcanna. (date of application – 01.12.2024).
5. I.M. Melnychuk. Peculiarities of using interactive games in higher education. UDC: 37.02:372. 12p.
6. Educational crossword puzzle: how and where to make it (title from the screen). url : https://www.eduneo.ru/uchebnyj-krossvord-kak-i-gde-sostavit/. (date of application – 01.12.2024).
7. Classic crosswords online. (title from the screen). URL: https://onlinetestpad.com/ua/crosswords/classic. (date of application – 03.12.2020).
8. Online quiz "Erudit". (title from the screen). URL: https://source2016.ru/event/. (date of application – 03.12.2024).
9. World history quiz (title from the screen). URL:https://play.google.com/store/apps/details?id=com.kalinovskiy.historyquiz.(date of application – 03.12.2024).
10. Crosswords on Russian (title) with screen). URL : https://play.google.com/store/apps/details?id=com.appspot.orium_blog.cr ossword. (date of application – 03.12.2024).
11. List of Android versions (title from the screen). URL: https://uk.wiipdia. org/wiki/СпыAndroid_versions_soc. (date of application – 07.12.2024).

12. Android Architecture (screenshot). URL: https://studbooks.net /2052989/informatika/arhitektura_android. (date of access – 08.12.2024).

13. Android Developers Blog (title from the screen). URL: https://android developers.googlblog.com/2020/05/android-studio-4.html. (accessed on 08.12.2024)

14. Visual Studio documentation (title from the screen). URL: https://docs. microsoft.com/ru-ru/visualstudio/windows/?view=vs-2019&preserve-view=true. (access date – 08.12.2024).

15. IDEF0 methodology. (title from the screen). URL: https://studme.org. /87184/economistika/meteorology_idef0/. (date of application – 10.12.2024).

16. Representation of data using the "essence-connection" model. (title from the screen). URL:http://www.mstu.edu.ru/study/materials/zelenkov/ch_2_1.html. (date of access – 09.10.2024).

17. Representation of data using the "essence-connection" model. (title from the screen). URL:http://www.mstu.edu.ru/study/materials/zelenkov/ch_2_1.html. (date of access – 06.12.2024).

18. Y. Leshchyshyn, L. Scherbak, O. Nazarevych, V. Gotovych, P. Tymkiv and G. Shymchuk, «Multicomponent Model of the Heart Rate Variability Change-point,» 2019 IEEE XVth International Conference on the Perspective Technologies and Methods in MEMS Design (MEMSTECH), Polyana, Ukraine, 2019, pp. 110-113, doi: 10.1109/MEMSTECH.2019.8817379

19. Lytvynenko, S. Lupenko, O. Nazarevych, G. Shymchuk and V. Hotovych, «Mathematical model of gas consumption process in the form of cyclic random process,» 2021 IEEE 16th International Conference on Computer Sciences and Information Technologies (CSIT), LVIV, Ukraine, 2021, pp. 232-235, doi: 10.1109/CSIT52700.2021.9648621

20. Bodnarchuk, I., Kunanets, N., Martsenko, S., Matsiuk, O., Matsiuk, A., Tkachuk, R., Shymchuk, H.: Information system for visual analyzer disease diagnostics. CEUR Workshop Proceedings 2488, pp. 43-56 (2019).

21. Шимчук Г. В. Дослідження методів захисту відомих хмарних платформ : кваліфікаційна робота освітнього рівня „Магістр“ „125 – Кібербезпека“ / Г. В. Шимчук. – Тернопіль : ТНТУ, 2022. – 74 с.

22. Методичні вказівки розроблені у відповідності з навчальним планом для студентів освітнього рівня бакалавр спеціальності 126 «Інформаційні системи та технології» / Уклад.: О. Б. Назаревич, Г. В. Шимчук, Н. М. Шведа. – Тернопіль : ТНТУ 2020. – 22 с.

23. V. Kozlovskyi, Y. Balanyuk, H. Martyniuk, O. Nazarevych, L. Scherbak and G. Shymchuk, «Information Technology for Estimating City Gas Consumption During the Year,» 2022 International Conference on Smart Information Systems and Technologies (SIST), Nur-Sultan, Kazakhstan, 2022, pp. 1-4, doi: 10.1109/SIST54437.2022.9945786.