Information Technology for the Operational Processing of Military Content for Commanders of Tactical Army Units

Vitaliy Danylyk
Information Systems and Networks Department, Lviv Polytechnic National University, 79013
E-mail: Vitalii.Danylyk.msaad.2021@lpnu.ua
ORCID iD: https://orcid.org/0000-0001-5928-7235

Victoria Vysotska
Information Systems and Networks Department, Lviv Polytechnic National University, 79013
E-mail: Victoria.A.Vysotska@lpnu.ua
ORCID iD: https://orcid.org/0000-0001-6417-3689

Vasyl Andrunyk
Information Systems and Networks Department, Lviv Polytechnic National University, 79013
E-mail: Vasyl.A.Andrunyk@lpnu.ua
ORCID iD: https://orcid.org/0000-0003-0697-7384

Dmytro Uhryn
Computer Science Department, Yuriy Fedkovych Chernivtsi National University, Chernivtsi, Ukraine
E-mail: d.ugryn@chnu.edu.ua
ORCID iD: https://orcid.org/ 0000-0003-4858-4511

Yuriy Ushenko*
Computer Science Department, Yuriy Fedkovych Chernivtsi National University, Chernivtsi, Ukraine
E-mail: y.ushenko@chnu.edu.ua
ORCID iD: https://orcid.org/0000-0003-1767-1882
*Corresponding author

Abstract: In the modern world, the military sphere occupies a very high place in the life of the country. At the same time, this area needs quick and accurate solutions. This decision can greatly affect the unfolding of events on the battlefield and indicate that they must be used carefully, using all possible means. During the war, the speed and importance of decisions are very important, and we note that the relevance of this topic is growing sharply. The purpose of the work is to create a comprehensive information system that facilitates the work of commanders of tactical units, which organizes the visualization and classification of aerial objects in real-time, the classification of objects for radio-technical intelligence, the structuring of military information and facilitates the perception of military information. The object of research/development is a phenomenon that creates a problematic problem, has the presence of slowing factors in the process of command and control, using teams of tactical links, which can slow down decision-making, as well as affect their correctness. The research/development aims to address emerging bottlenecks in the command-and-control process performed by tactical link teams, providing improved visualization, analysis and work with military data. The result of the work is an information system for processing military data to help commanders of tactical units. This system significantly improves on known officer assistance tools, although it includes a set of programs that have been used in parallel on an as-needed basis. Using modern information technologies and ease of use, the system covers problems that may arise for commanders. Also, each program included in the complex information system has its degree of innovation. The information system for structuring military information is distinguished by the possibility of use on any device. The information system for the visualization and clustering of aerial objects and the information system for the classification of objects for radio technical intelligence are distinguished by their component nature. This means that the application can use sources of input information and provides an API to use other processing information. Regarding the information system for integration into information materials, largely unknown terms and abbreviations are defined, so such solutions,

cannot integrate the required data into real documents. Therefore, using this comprehensive information system, the command of tactical units will have the opportunity to improve the quality and achieve the command-and-control process.

Index Terms: Military Data, Commander of Tactical Units, Information Structuring, Natural Language Processing, Content Classification.

1. Introduction

Command and control processes undoubtedly have one of the most important values in the military sphere [1]. A lot of time is devoted to the training of commanders because it directly affects the results achieved. Command and control procedures are not clear. Much depends directly on the commander and the actions; he takes in extreme moments. The role and place of the commander in the operation can often change depending on the specific phase of the operation. As the operation progresses, the commander's freedom of action decreases, while external constraints increase. From being the primary player on the ground, it becomes a provider of support to other civilian or local players. At the same time, regardless of the phase of the operation, the basics of command remain unchanged; in particular, it is necessary to adapt the military commander to the environment in which he operates and give meaning to his actions [2]. Command and control on the battlefield are a constant balancing act [3]. The need for detailed planning must be balanced with the need for quick and decisive action. The command-and-control process helps the commander and his staff achieve this balance as they apply their experience, judgment, and creativity to a common goal. The tactical commander's role is to protect life, property, and the environment by ensuring rapid and effective action that saves lives and minimizes harm. They work between strategic and operational levels of command. Tactical commanders are responsible for interpreting strategic direction (where the strategic-level command is used) and developing and coordinating the tactical plan.

The position of commander involves processing a large amount of information. Any decisions must be carefully considered, supported by facts and rational predictions of their potential outcomes. The commander must take into account all factors and be guided by already thought-out and described tactics and strategies [4-6]. Also, one of the main factors of good command is speed. So, the commander's task is quick and thoughtful command and control. But finding the right solution quickly can be a very difficult task. After all, to do this, you need to quickly analyze all the information in general, and sometimes this is impossible. In today's world, the latest technologies are developing very quickly. Many services make work easier for many people. Accordingly, they should be actively used in the military sphere. The latest information systems will be able to cover possible problems and help officers make quick and rational decisions.

In the process of work, commanders can face 3 types of problems:

- lack of knowledge - it can be difficult for a person to retain all the accumulated knowledge due to the amount; this can cause a situation when the ideal solution to the problem was already described earlier, but the person does not remember it;
- slow information analysis – the analyzed information should be accessible as soon as possible and in a complete form, but the analysis process can be quite long;
- incomprehensibility of information materials - in the military sphere there is a large amount of documentation, which without a doubt contains many terms and abbreviations, ignorance of which affects the comprehensibility of the information presented.

A single program cannot cover these challenges that a tactical unit commander may face. Accordingly, it will be appropriate to create a set of programs that will cover their part of the problems and thus help officers to make quick and well-considered decisions. To cover the problems mentioned above, it is necessary to create a complex of information systems that will collectively represent a complex system of processing military data to help commanders of tactical units. The task of the work is the development of this system. Since the target system is a complex of information systems, the task of the work is the development of all these systems. Information systems included in the complex system can be classified according to the problems they are designed to solve: knowledge and analysis. So, to cover possible problems, you need to develop 4 independent programs:

- A program for structuring military information. This information system belongs to the category of knowledge and is intended for the organization of various information materials. It will make it possible to conveniently and quickly use and search for information in a single knowledge base. This will enable officers to quickly find the necessary information and use it for their purposes. Also, this system is very well suited for educational purposes because the information will be well structured.
- A program for visualization and classification of aerial objects in real-time. This information system is included in the category of analysis and is intended for convenient visualization and classification of aerial objects in real-time. This will allow officers to quickly react to various events because the information will be analyzed and accordingly visualized automatically.
- A program for the classification of objects for radio technical intelligence. This information system is included

in the category of analysis and is intended for the classification of objects for radio technical intelligence. Commanders need to know what resources the enemy is using. Since on the battlefield, there will very likely be means of radio-technical intelligence, the electromagnetic waves of which are visible at a long distance, using this program, it will be possible to find out exactly the resources used by the enemy.

- A program for integrating definitions of potentially unknown terms and abbreviations into informational materials. This information system belongs to the category of knowledge and is intended for processing documents to integrate definitions of potentially unknown terms and abbreviations into them. This will allow officers to use any literature without any problems because all terms and abbreviations will be known.

In the modern world, the military sphere occupies a very high place in the life of the country. At the same time, this area needs quick and accurate solutions. These decisions can greatly affect the unfolding of events on the battlefield, which means that they must be taken carefully, using all possible means. In the period of war, the speed and attentiveness of decisions are very important, which means that the relevance of this topic is growing sharply.

The goal of the work is to create a comprehensive information system that will facilitate the work of commanders of tactical units, which organizes visualization and classification of air objects in real-time, classification of objects for radio technical intelligence, structuring of military information and facilitating the perception of military information. The object of research/development is a phenomenon that creates a problematic situation, the presence of slowing factors in the process of command and control carried out by commanders of tactical units, which can slow down decision-making and also affect their correctness. The subject of research/development is to solve the problems of the appearance of slowing factors in the process of command and control carried out by commanders of tactical units, using improving visualization, analysis and work with military data. To develop a complex information system for processing military data to help commanders of tactical units, 4 tasks must be solved:

- Develop a program for structuring military information.
- Develop a program for the visualization and classification of aerial objects in real-time.
- Develop a program for the classification of objects for radio technical intelligence.
- Develop a program to integrate definitions of potentially unknown terms and abbreviations into informational materials.
- Combine the developed programs to form a comprehensive information system.

To develop an information system for processing military data to help commanders of tactical units, the following tasks must be completed:

- Analyze some literary sources for studying the subject area.
- Analyze the available means of solving problems.
- To carry out a system analysis to separate the ultimate goal of the project and sub-goals, to achieve an acceptable level of decomposition and to define the processes in the system.
- Build a conceptual model using UML notation.
- To carry out practical implementation of the information system for processing military data to help commanders of tactical units.
- Conduct testing and implement the project.

Innovativeness (novelty) of work results. The result of the work is an information system for processing military data to help commanders of tactical units. This system significantly improves known means of helping officers, because it includes a set of programs that are used in parallel depending on the need. Using modern information technologies and ease of use, the system covers problems that may arise for commanders. Also, each program included in the complex information system has its degree of innovation. The information system for structuring military information is distinguished by the ability to use it on any device. The information system for the visualization and clustering of aerial objects and the information system for the classification of objects for radio technical intelligence are distinguished by their component nature. This means that the application can use different sources of input information and provides an API to use the processed information. As for the information system for integrating definitions of potentially unknown terms and abbreviations into information materials, there are similar solutions, but they cannot integrate the necessary data into real documents. Therefore, using this comprehensive information system, commanders of tactical units will be able to improve the quality and speed up the process of command and control.

2. Analysis of Recent Research and Publications

Ground military actions, initiated at the strategic level and planned at the operational level, are developed within a global crisis management system that combines political, diplomatic, defense, cultural, economic and communication aspects. In this context, all civil and military players must act in synergy and coherence. Therefore, all actions must be part of a logical process aimed at achieving the political goal set before the operation. Therefore, the military commander

must not only take into account the presence of other players and their actions on the battlefield but also seek optimal coordination with them. Therefore, it is he who must correctly position himself about these players, ensuring that his actions, starting from the planning phase, are fully consistent with the goals of non-military operations [1]. It is important to understand what command and control is a command-and-control system, and what these terms mean individually in a military context [2]. Several authors rephrase the definition, and they are all relatively the same.

Command. The authority that a commander in the Armed Forces legitimately exercises over subordinates based on rank or appointment. Command includes the authority and responsibility for the effective use of available resources and the planning, organization, direction, coordination, and control of military forces to accomplish assigned tasks.

Control. Authority that may be less than the commander's full command over a portion of the activities of subordinates or other organizations.

Command and control. The exercise of authority and direction by the duly designated commander over detached and attached forces during the execution of the mission [3]. Command and control functions are accomplished through the organization of personnel, equipment, communications, facilities, and procedures used by the commander to plan, direct, coordinate, and control forces and operations during the mission. Command and control separately refer to the authority of commanders over subordinates. It involves the planning, organization, direction, coordination and control of the forces under their command or control. A command-and-control system includes personnel, equipment, communications, facilities, and procedures that contribute to the ability of commanders to command or control. Command and control together are the processes by which commanders use personnel, equipment, communications, facilities, and procedures (command and control systems) to plan, organize, direct, coordinate, and control (command or control) their subordinates. The last definition above describes command and control as a process. The Lawson loop is a commonly accepted model of the command-and-control process. Developed by Joel S. Lawson, it was applied to these processes from the Napoleonic era until the 1990s. The model consists of five functions, namely: feeling; process; comparison; solution; and action.

Feeling. The sensing function collects data from the environment. The commander uses all available means, including intelligence gathered by soldiers, radars, and unmanned aerial vehicles. The environment includes friendly and enemy forces, terrain, weather, and other information valuable to commanders.

Process. The process function aggregates and correlates the data collected by the sense function. Although Lawson's loop function called "process" is a common term in the information management community, the military community refers to this function as data fusion. This is the process of studying all sources of information and intelligence to obtain a complete assessment of the activity.

Comparison. The commander takes the information just processed in the process function and compares the current state of the environment to the desired state of the environment. The current state can indicate enemy strengths and weaknesses, locations, and intentions. The commander's staff will likely develop courses of action to move the current environmental state to the desired state.

Solution. After the comparison function, commanders choose the courses of action presented by their staff. This function may seem like a small part of a larger process, but it is a thorough process by which the commander examines and compares available options for viability and effectiveness. In some cases, the commander may reject all or part of the courses of action, which requires additional planning by the headquarters. In addition, commanders will report the decision and follow-up to higher headquarters: this may also influence deviations/changes in the course of action.

Action. The action function is a manifestation of the commander's decision. The commander gives orders to influence the current state of the environment.

Lawson's loop classifies the components of the commander's work quite well, and accordingly, the information systems developed to assist commanders cover one or more categories of this cycle. The information system developed in this work covers the functions of sensing and solving. In particular, Information systems belonging to the group of analysis can be attributed to the function of feeling, and information systems belonging to the group of knowledge can be attributed to the function of decision. As of 2022, we can single out several popular and successful software solutions that have been operating for a considerable period and that have been able to prove themselves as systems that are worth checking out. These systems are not necessarily of a military nature, but their purpose is to overcome the same problems that arise in the process of command and control carried out by commanders of tactical units.

Fandom is an American wiki service, also known as Wikia. Fandom uses MediaWiki, an open-source wiki software. Fandom has a global audience of over 200 million monthly and over 400,000 communities. It's a source of entertainment trusted by fans with over 50 million pages. A wiki is like a database, collaboratively developed by a community of users, where they can edit and add content. Fandom wikis consist of online encyclopedias, each specializing in a specific topic. This service can be used from any device that has access to the Internet using a web browser. The main purpose of a Wiki in the Fandom community is to highlight information and discussion about a particular topic at a much deeper level than what can be found anywhere else on the Internet. The most popular wikis on the Fandom service are constantly updated according to user information. This is a community website. Anyone can edit an existing article or create a news, wiki, or Fandom post. Everyone can work together to make Fandom a better place tomorrow and expand that knowledge base.

Klassify, Data Classification Suite is a data classification platform that helps organizations communicate data classification policies to their users and forces them to apply the appropriate classification to the data they create, share, and store based on data sensitivity. Data classification with Klassify eases organizations' compliance obligations with a variety of regulatory requirements, including PCI DSS, EU GDPR, HIPAA, DPA, CCPA and many others, and provides detailed metadata and contextual information to improve information management and organizational efficiency to

manage unstructured data Klassify's data classification package helps build a strong foundation for data security by forcing users to apply classification labels and corresponding labels to documents and emails, clearly defining the sensitivity of the information.

Dataedo is one of the most popular data dictionary tools. In 2020, it was nominated by Gartner Peer Insights Customers' Choice as the go-to metadata management solution. The software allows users to navigate between millions of documents and exchange data in HTML format. Dataedo helps find meaning in data by adding insights to datasets that users are familiar with. Finding data is easy with Dataedo. Users can add relationships between datasets and tables to provide the reader with a better understanding of the data. Organized data means continuity of business processes. With different teams and departments owning different sets of data, data organization becomes imperative. Dataedo helps manage the complexity of business operations by allowing users to create segmented data profiles for each department/team to help organize large data sets.

Basic indicators that make it possible to evaluate and compare the advantages and disadvantages of the use of the closest analogues [7-17]:

- Platform – quantitative assessment;
- Operating system - quantitative assessment;
- Programming language - evaluation of advantages (better/good/worse);
- Necessary resources – evaluation of benefits (more/less);
- Interoperability – Boolean evaluation (yes/no);
- Mobility – Boolean evaluation (yes/no);
- Scalability – Boolean evaluation (yes/no);
- Interaction with the user - quantitative assessment;
- Functionality - quantitative assessment;
- Suitability for use - quantitative assessment;
- Reliability - quantitative evaluation;
- Productivity - quantitative assessment;
- Usability - quantitative assessment.

A scale that will make it possible to quantitatively evaluate and compare the advantages and disadvantages of using the closest analogues:

- 5 – best-selected software, operating system, very user-friendly interaction, full functionality, full usability, reliability, performance and operability.
- 4 – well-chosen software and operating system, but only for a certain range of users, partial functionality, full usability, reliability, performance and operability.
- 3 – well-chosen software and operating system, but only for a certain range of users and applications in a similar field (not for forestry, but for agriculture), partial usability, reliability, performance and operability.
- 2 – well-chosen software and operating system, but only for a certain range of users and applications in a similar field, poor usability, reliability, performance and operability.
- 1 – poorly selected software and operating system, limited use, poor usability, reliability, performance and operability.

Table 1. Comparison table of characteristics for the project and the closest analogues

Characteristics	Project	Closest analogues		
		Fandom	Klassify	Dataedo
Platform	4	5	4	5
Operating System	5	5	5	5
Programming language	better	worse	better	worse
Necessary resources	less	more	more	more
Interoperability	yes	no	no	no
Mobility	no	yes	yes	yes
Scalability	yes	yes	no	yes
Interaction with the user	5	4	4	3
Functionality	5	4	4	4
Suitability for use	5	5	4	3
Reliability	5	5	4	4
Productivity	5	5	4	5
Usability	4	5	5	5

There is no full-fledged integrated information system that would process military data to help commanders of tactical units. After analyzing the closest analogues and comparing them with the target product, you can notice quite a few differences, namely [7-14]:

- Some analyzed programs are quite progressive and have a lot of functionality. If you need speed in work, a long setup and deployment of information can be a problem. Accordingly, it will be a good approach to determine the basic needs of the user and provide only the functionality that is needed. This is the approach of the targeted information system.
- Some analyzed services work online and can be accessed from anywhere in the world. In the military sphere, this is not very good. Accordingly, the solution to this problem is the implementation of all information systems in the form of software or the possibility of their deployment in a local network. This is how the information system that is being developed works. Some programs will be installed on the computer and some will be opened in a browser on the local network.
- None of the analyzed programs can work with data that comes in real-time. In the closest analogues of programs for data analysis, you need to enter updated information manually or using import tools. For quick analysis, automatic data provision is a very important advantage of the program. The information system for the visualization and classification of aerial objects works in real-time. By connecting it with aerial object tracking devices, data analysis and visualization will take place instantly.
- An advantage of the analyzed programs may be their cross-platform nature. Most of the integrated information system programs being developed will be deployed on a specific platform, but the programs that will potentially be used by a wide audience will be cross-platform. This will enable users in the local network to use information systems on any device.
- The software that was analyzed uses the approach of one unique program. That is, to have some advantages, you need to use the corresponding programs in parallel with others. This can become a problem when the user needs to switch from software that he has been using for a long time to a new program that he still needs to learn. It will also be inconvenient to use more than one program in parallel to achieve the corresponding goal. One of the programs of the complex system is designed to solve these problems. An information system for integrating definitions of potentially unknown terms and abbreviations into information materials will generate input material with additional data. Thus, it will be possible to use it in old and convenient programs with additional necessary information.

Comparing the target information system with the closest analogues, it can be concluded that it is both relevant and important for commanders of tactical units. Using this system, commanders are likely to speed up the process of command and control and increase the probability of making the right and rational decision. Using this system, users will feel comfortable because it will have only the necessary functionality and a convenient graphic interface. Different programs of the system will be responsible only for a specific field of activity and, by it, will have their advantages: cross-platform, working with data in real-time, protected work in a local network, and integration of information into the materials of already known and popular programs.

3. Goal Formulation and Task Setting

System analysis is a process of collecting and interpreting facts, identifying problems, and decomposing the system into its components [4-14]. System analysis is conducted to study the system or its parts to determine its goals. It is a problem-solving technique that improves the system and ensures that all components of the system work efficiently to achieve their purpose. This information can inform decision-making or improve the efficient use of time, money and resources in business and life. Since the information system being developed is complex, that is, it includes several programs, it is necessary to model the requirements both for the complex information system in general and for each program that makes up the system separately. A complex information system should consist of programs that are united by the appropriate purpose. These programs should add additional functionality to the system and be independent. Thus, removing or adding new programs to the system will not have any effect on other system programs. The software architecture should ensure the scalability and expansion of the program [15]. Adding additional functionality should be done by adding additional modules without significant modernization of already existing modules. The architecture of programs should follow the generally accepted principles of MVC architecture [16]. The system should allow users to have the following capabilities:

- Organization of information materials – user management of information materials.
- Use of informational materials – viewing of informational materials by the user.
- Classification of aerial objects – obtaining information about aerial objects in real-time.
- Visualization of aerial objects - the user must be able to visually view the coordinates of aerial objects.
- Classification of objects for radio technical intelligence - obtaining information about objects for radio technical intelligence.

- Adding terms and abbreviations to the dictionary - the user should be able to add terms and abbreviations to the dictionary for further use.
- Combining dictionary and information material – the user should be able to combine the dictionary with information material, resulting in a copy of the input information material with additional data.

Software requirements.

- all servers involved in the system are operated under the pressure of operating systems of the MS Windows family unless otherwise specified in the technical specifications;
- all servers and other computing equipment of the system are combined into a single centralized management and monitoring system;
- all servers work in a local network to ensure data security;

Hardware requirements.

- System hardware and system programs must have UkrSEPRO compliance certificates.
- The composition of the hardware must meet the requirements for the availability and protection of information.

The main language of the system interface and all programs included in this complex information system is Ukrainian. Requirements for a software platform and code for structuring military information.

- The application must be developed using HTML + CSS + JS + PHP + JQuery + CKEditor.
- The text encoding to be used on all pages of all language versions is UTF-8.
- Use of the program must be available via the HTTPS protocol (with cryptographic protection of data exchange between the website and the workplace).
- The software must function 24 hours a day, 7 days a week, 365 days a year (provided that the hardware functions without fail).
- Software must be able to request interruptions for routine maintenance and/or backup of information content.

The program should provide the ability to back up and restore information arrays through the administrator's web interface. The application must be protected against XSS, as well as other means of interfering with the continuous operation of its software platform.
The program should allow users to perform the following operations:

- Access to information – the user must be able to view the information.
- Searching for content – the user enters a keyword to search for the desired content.
- The division into topics – receiving information based on the relevant topic.
- Tree-like structure of pages – pages of information material should form a tree-like structure.
- Information management – a user with appropriate rights must be able to manage information.

Software requirements.

- the client part of the program should not be installed on personal computers.
- a user with administrator rights must have a graphical interface.

Software platform and code requirements for real-time visualization and classification of aerial objects.

- The application should be developed using HTML + CSS + JS + NodeJS + ElectronJS + BrainJS.
- Text encoding is to be used on all pages of all language versions – UTF-8.
- The software must function 24 hours a day, 7 days a week, 365 days a year (provided that the hardware functions without fail).
- Software should not require interruptions for routine maintenance and/or backup of information content.

 The program should allow users to perform the following operations:

- Visualization of aerial objects – the user should be able to use a user-friendly graphical interface to view the data visualization.
- Classification of aerial objects – obtaining information about the classes of aerial objects in a convenient form.
- Real-time operation – Receives input data automatically for instant processing.

Also, the program must be installed on personal computers. Requirements for the software platform and code for the classification of objects for radio technical intelligence.

- The application must be developed using HTML + CSS + JS + JQuery.
- Text encoding is to be used on all pages of all language versions – UTF-8.
- Use of the program must be available via the HTTPS protocol (with cryptographic protection of data exchange between the website and the workplace).
- The software must function 24 hours a day, 7 days a week, 365 days a year (provided that the hardware functions without fail).
- Software should not require interruptions for routine maintenance or backup of information content.

The program should allow users to perform the following operations:

- Parameter management – the user must be able to enter data that will be parameters for the classifier.
- Classification of objects for radio technical intelligence – obtaining information about the probability of the class to which the object with input data belongs.
- Instant update – update the result after each change of the input parameter.

The client part of the program should not be installed on personal computers.

Requirements for the software platform and code for the integration of definitions of potentially unknown terms and abbreviations into informational materials.

- The program must be developed using Python + Tkinter + PyMuPDF.
- The software must function 24 hours a day, 7 days a week, 365 days a year (provided that the hardware functions without fail).
- Software should not require interruptions for routine maintenance and/or backup of information content.

The program should allow users to perform the following operations:

- Selection of dictionary and informational PDF file – the user must be able to select both the informational PDF file and the dictionary before integration.
- CSV dictionary is the user should be able to use CSV as a database of all terms, abbreviations and their definitions.
- Notes as definitions – integration into PDF of notes with definitions of terms and abbreviations that appear when displayed in PDF viewer programs.
- Generating input extension file – saving the result as an information file with the same extension as the input information file.
- Saving the last used dictionary - the next time you enter the program, the last used dictionary will be selected automatically.

4. Presentation of the Main Material

Before developing an information system, you need to fully understand the target product, its place in the environment and all the functions it will cover. For these purposes, developers create a conceptual model of the system before the direct process of its development [17]. Thus, during the development process, all goals and the final result that should be obtained will be known. Unified Modeling Language (UML) is a standardized modelling language. It helps software developers visualize, build, and document new software systems. UML is used to create static structure diagrams based on a variety of engineering techniques that have proven effective in creating complex systems. Historically, UML was used to model software systems, but it was not limited to software developers. Today, UML diagrams are also used for process and project management. Since the information system being developed is complex, appropriate diagrams were developed both for the system as a whole and for each program that includes the system separately. A use case diagram was used to summarize the details of system users and their interaction with the system (Fig. 1). It visualizes both the general system and the programs it contains. Thus, this diagram contains options for the users of the system under development.

The complex information system for processing military data to help commanders of tactical units contains a certain number of programs. These programs are not necessarily of the same type and are not necessarily independent. They can use external services and be a web service themselves. Thus, the operation of some programs will be more complicated than a simple program for a personal computer. A deployment diagram was used to display computing nodes during the operation of programs, components, and objects executed on these nodes (Fig. 2). Thus, it is possible to see exactly how the information system is deployed, and how the components interact with each other and with external objects. The diagrams shown in Fig. 1 and Fig. 2 describe the system in general. For a more detailed description of the processes, it is necessary to describe each program that is in the system being developed. The program for structuring military information involves the work of 2 users. Accordingly, two diagrams are required for all display processes occurring in the program: an activity diagram for the visitor (Fig. 3) and an activity diagram for the administrator (Fig. 4).
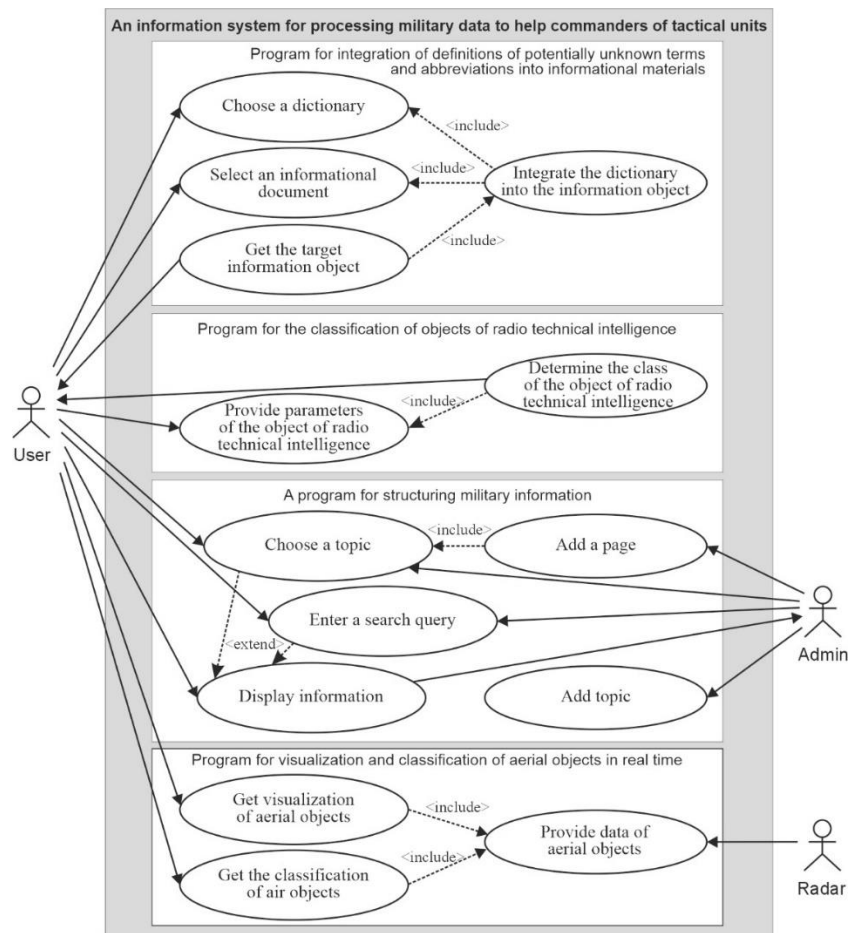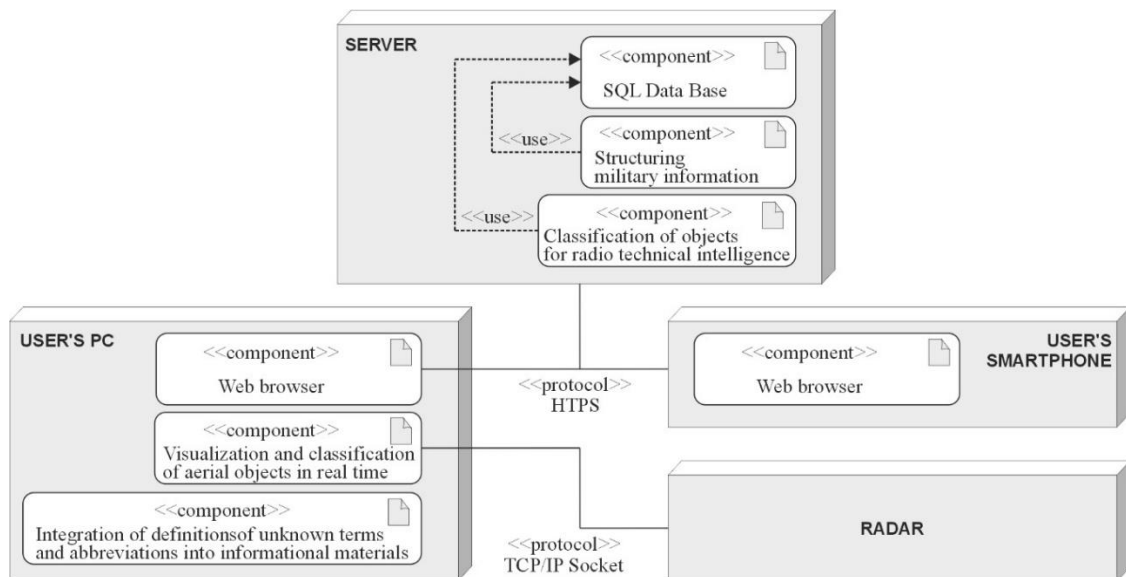
Fig.1. Diagram of options for using the system

Fig.2. System deployment diagram

The administrator of this program is an advanced visitor. It has all the same capabilities as a simple user (Fig. 3) and also additional capabilities that are available only to this type of user (Fig. 4). The concept of this program includes four objects: user, administrator, program and database. A sequence diagram was used to display the interactions of these objects (Fig. 5). Also, this diagram shows object interactions ordered by time. The program for visualization and classification of aerial objects in real time involves the work of one user, but it interacts with an external source of information - radar. A state diagram was used to display the general operation of the program (Fig. 6a). This diagram contains all the states and transitions that the program contains. The concept of this program includes three objects: the user, the radar, and the program. The sequence diagram (Fig. 6b) shows the interactions of these objects. The program

for the classification of objects for radio technical intelligence involves the work of one user. It is independent and does not interact with external objects. A state diagram was used to display the general operation of the program (Fig. 7a). This diagram contains all the states and transitions that the program contains. The concept of this program includes two objects: the user and the program. The sequence diagram (Fig. 7b) shows the interactions of these objects. The program for integrating definitions of potentially unknown terms and abbreviations into informational materials requires the work of one user. It is independent and does not interact with external objects. The program being developed uses the files of other programs. In particular, PDFs for information files to which unknown terms and abbreviations from the dictionary will be added, and word files, in turn, have the CSV extension. To use this type of dictionary, you need to create it using other programs. A state diagram was used to display the general operation of the program (Fig. 8). This diagram contains all the states and transitions that the program contains.
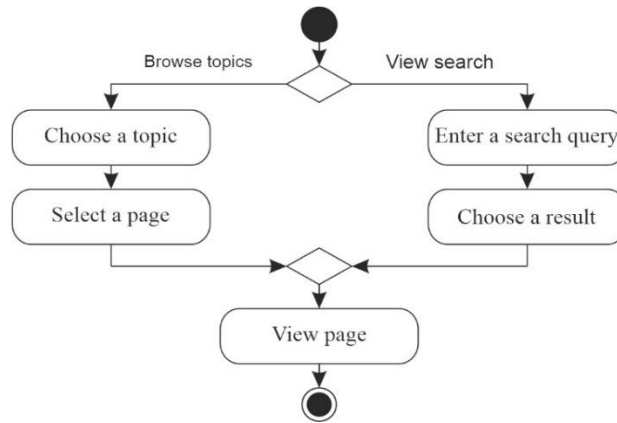
Fig.3. Activity diagram for the visitor of the program for structuring military information
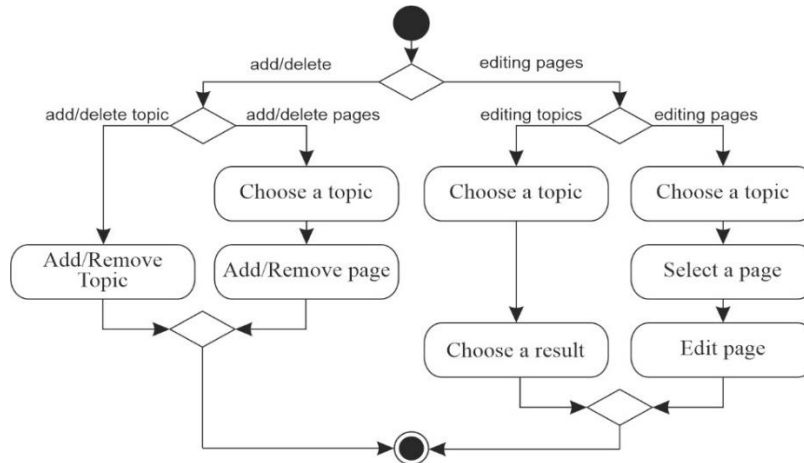
Fig.4. Activity diagram for structuring military information

The concept of this program includes two objects: the user and the program. The sequence diagram (Fig. 9) shows the interactions of these objects. The diagram does not show other programs that create an information file or create a dictionary since to work with the program to integrate definitions of potentially unknown terms and abbreviations into information materials, you need to have ready-made files for merging. They are used as input data.

Having modelled the processes that take place in the information system, an idea emerges of how the system should function. For its implementation, it is necessary to simulate the object model of the software itself. In an object-oriented approach, the focus is on capturing the structure and behaviour of information systems in small modules that combine both data and processes. The main goal of object-oriented design is to improve the quality and productivity of system analysis and design, making them more user-friendly [18]. In the analysis phase, object-oriented models are used to fill the gap between the problem and the solution. It works well in a situation where systems are constantly being designed, adapted and maintained. It identifies objects in a problem domain by classifying them based on data and behaviour. The complex information system being developed consists of several programs, which should be modelled using an object-oriented approach.
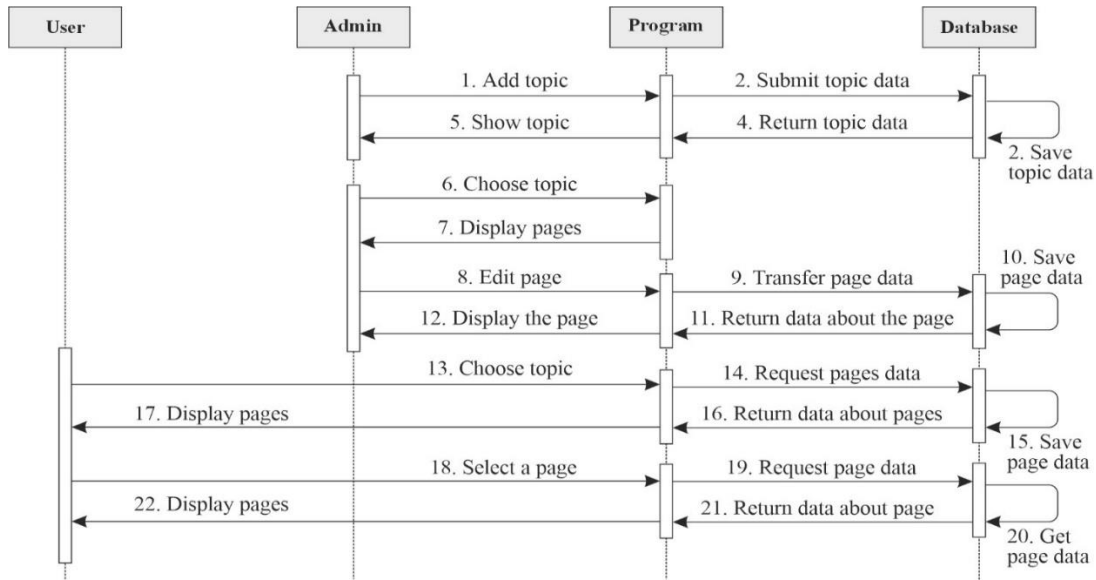
Fig.5. Sequence diagram of the program for structuring military information
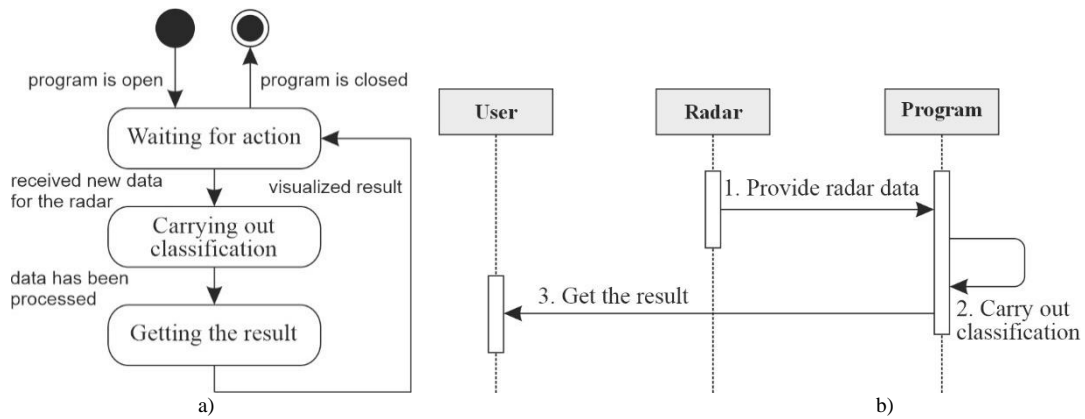

Fig.6. Diagram of a) states and b) sequence of the program for visualization and classification of aerial objects in real-time
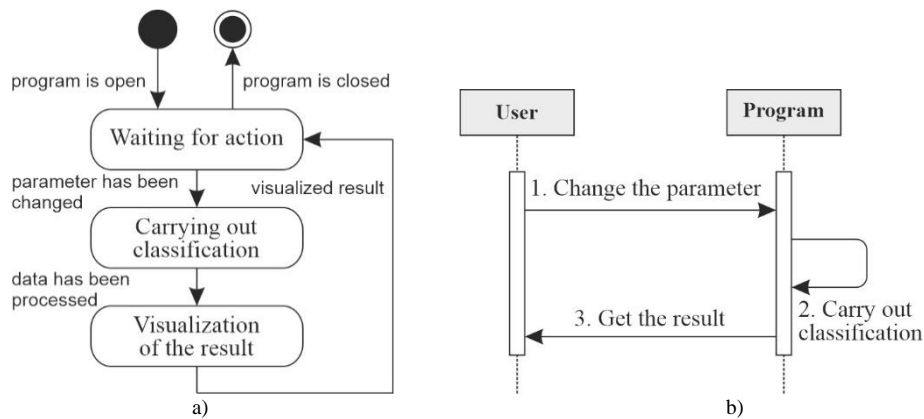

Fig.7. State diagram of the program for the classification of objects for radio technical intelligence

The program for structuring military information provides convenient page navigation, good visualization of information and hierarchical connection of pages. To solve these main problems, a corresponding class diagram was developed (Fig. 10). The IMenu and IViewer interfaces provide an opportunity to experiment with different menu and viewer implementations, which means that you can quickly choose the best implementation without significant intervention in the architecture. The presence of the children field in the Page class makes it possible to use a hierarchical connection of pages [19].
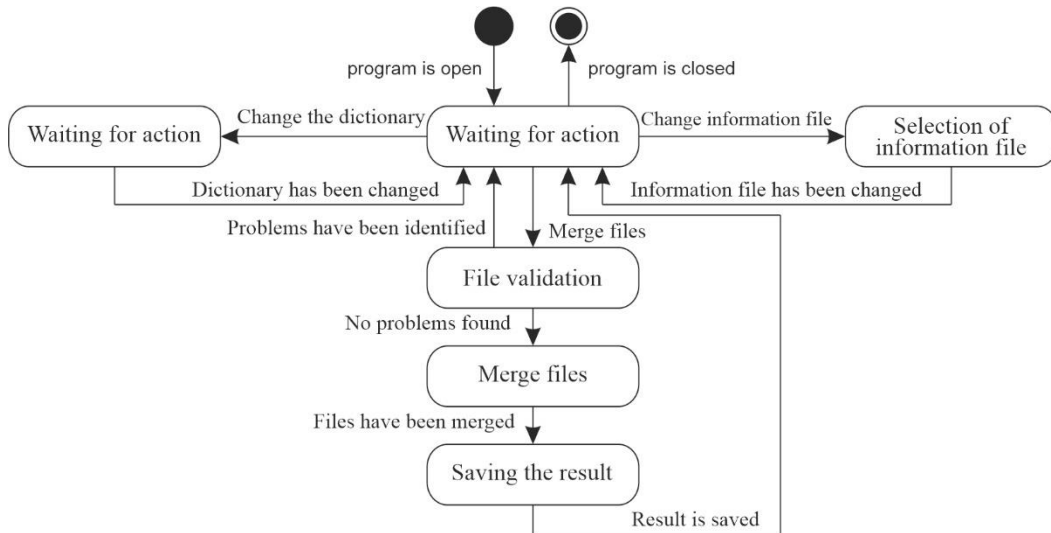
Fig.8. Diagram of program states for integration of definitions of potentially unknown terms and abbreviations into informational materials
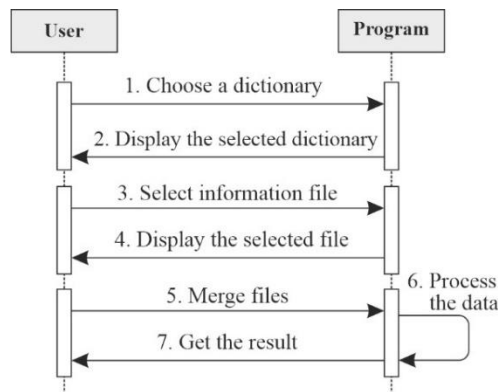
Fig.9. Sequence diagram of the program for integration of definitions of potentially unknown terms and abbreviations into informational materials
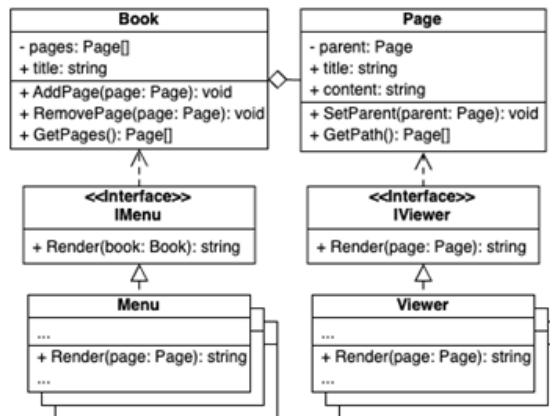
Fig.10. Class diagram of a program for structuring military information

The program for visualization and classification of aerial objects in real-time assumes the presence of a classifier that will classify objects. However, classification algorithms and data can change rapidly. To solve this main task, a corresponding class diagram was developed (Fig. 11). Using template classes, the architecture makes it possible to change the class that is responsible for the data of the object that is subject to classification and can potentially change. Thus, all that remains is to create a new class that implements the IClassifier interface of the appropriate type for processing new data models or new algorithms.
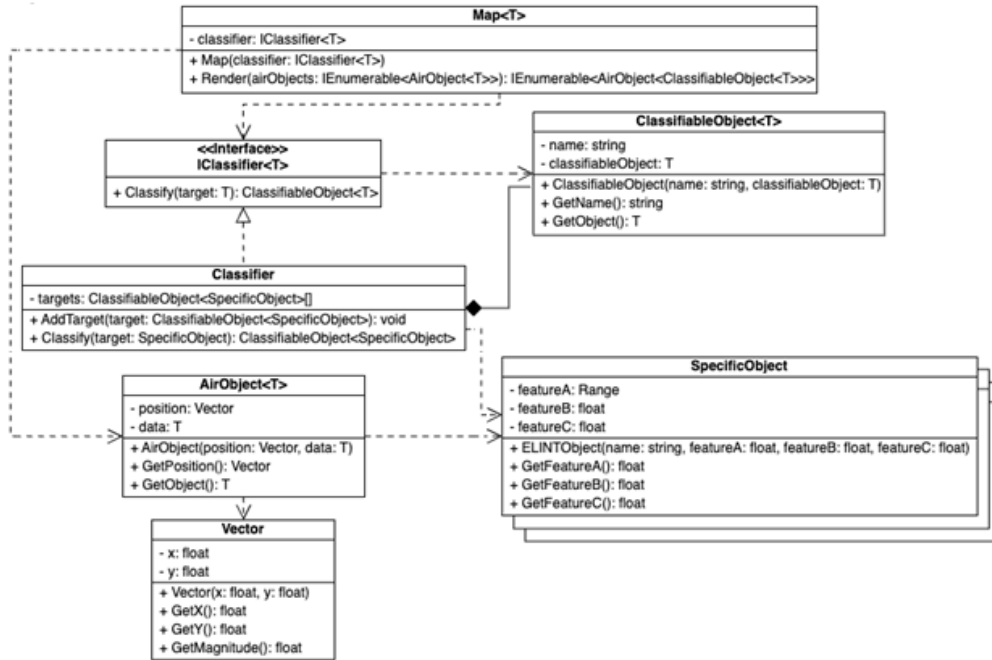
Fig.11. Class diagram of the program for visualization and classification of aerial objects in real-time

The program for classifying objects for radio technical intelligence assumes, as in the previous case, the presence of a classifier that will classify objects. The general scheme is very similar to the case of the program for the visualization and classification of aerial objects in real-time. To solve this main task, a corresponding class diagram was developed (Fig. 12). Although the main task is the same as in the previous case, the classifier is still different. It should return the probabilities of all classes, which means special objects, as shown in the diagram.
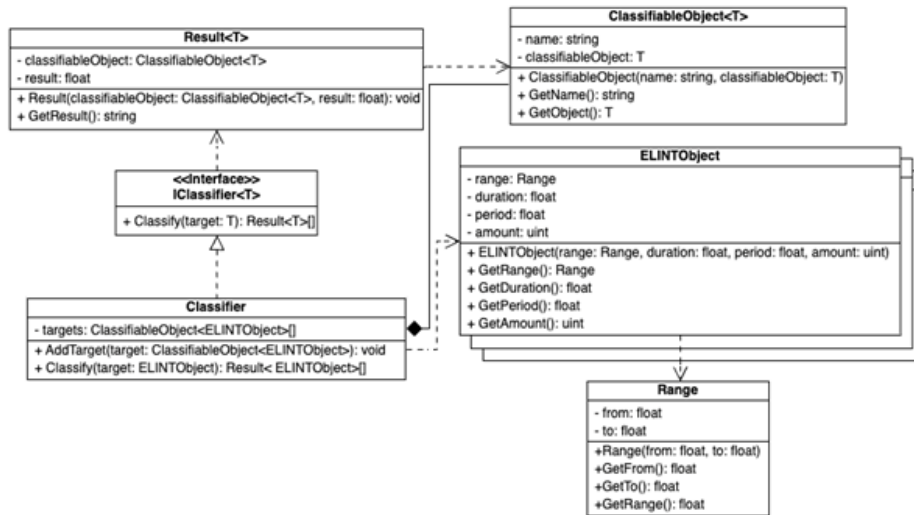
Fig.12. Class diagram of the program for classifying objects for radio technical intelligence

The program for integrating definitions of potentially unknown terms and abbreviations into informational materials involves working with informational files and dictionaries. These files can have different extensions, so each extension will have its implementation. To solve this main task, a corresponding class diagram was developed (Fig. 13). For good design, the architecture was designed based on interfaces. In this way, you can add different implementations of IResource and IDictionary interfaces, which makes it possible to expand the program for many other types of files in the future.

Software development refers to a set of computer science activities devoted to the process of creating, designing, deploying, and maintaining software. The software itself is a set of instructions or programs that tell a computer what to do.

Having defined the purpose, requirements and goals, analyzed analogues and performed a system analysis, it became fully known about the final target result. Using a conceptual model, there is a complete picture of the system, its functionality and users. The next step in the development of the project is the direct development of the graphical interface, algorithms and code writing.
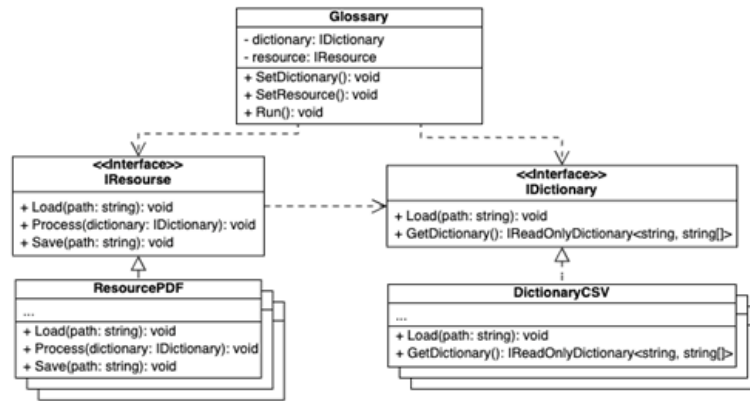
Fig.13. Class diagram of the program for integration of definitions of potentially unknown terms and abbreviations into informational materials

A comprehensive information system for processing military data to help commanders of tactical units includes several programs with different purposes but united by one goal. Accordingly, each of them has different requirements and different main functional features.

4.1. Classification Method

One of the features is data classification. Classification is the process of recognizing, understanding, and grouping ideas and objects into predetermined categories or "subgroups" [20]. Using pre-categorized training data sets, machine learning programs use a variety of algorithms to classify future data sets into categories. There are several basic classification methods: logistic regression, naive Bayes classifier, k-nearest neighbours, decision tree, and support vector machine.

Method of logistic regression. Logistic regression is a calculation used to predict a binary outcome: either something happens or it doesn't [21]. This can be represented as Yes/No, Satisfactory/Unsatisfactory, Alive/Dead, etc. The independent variables are analyzed to determine a binary outcome with outcomes belonging to one of two categories. Independent variables can be categorical or numerical, but the dependent variable is always categorical. This can be used to calculate the probability that a word has a positive or negative value (0, 1, or on a scale in between). Or, it can be used to identify an object in a photo (trees, flowers, grass, etc.), with each object given a probability between 0 and 1.

Naive Bayes classifier method. Naive Bayes calculates the probability of whether a data point belongs to a certain category or not [22]. During text analysis, it can be used to classify words or phrases according to a preset "tag" (classification) or not. Naive Bayes is a family of probabilistic algorithms that take advantage of probability theory and Bayes theorem to predict the tag of text (such as news or customer feedback). They are probabilistic, meaning they calculate the probability of each tag for a given text and then output the tag with the highest probability value. They obtain these probabilities using Bayes' theorem, which describes the probability of a feature based on prior knowledge of the conditions that may be associated with that feature.

The method of k-nearest neighbours. K-Nearest Neighbors (k-NN) is a pattern recognition algorithm that uses training data sets to find k nearest relatives in future examples [23]. When k-NN is used in classification, the computation of placing the data into the nearest neighbour category is performed. If k = 1, then it will be placed in the class closest to 1. K is classified by polling the set of its neighbours. Using this algorithm, you can operate on many characteristics. The number of characteristics is responsible for the number of dimensions of the space in which the classified objects will be placed. Using vectors in a multidimensional space, you can easily calculate the distance between points and use them to find the nearest neighbours.

Decision tree method. A decision tree is a supervised learning algorithm that is ideal for classification problems because it can order classes at a precise level [24]. It works like a flowchart, dividing the data points into two similar categories at a time from the "trunk of the tree" to the "branches" and "leaves", where the categories become more finitely similar. This creates categories within categories, allowing for organic classification with limited human oversight. A random forest algorithm is an extension of the decision tree in which many decision trees are first created with training data, and then new data is inserted into one of the trees as a "random forest". Essentially, it averages the data to relate it to the nearest tree on the data scale. Random forest models are useful because they solve the decision tree problem of unnecessarily "glueing" data points into a category.

The method of the support vector machine. A support vector machine (SVM) uses algorithms to train and classify data within degrees of polarity, outputting it to X/Y prediction boundaries [25]. SVM assigns a hyperplane that best separates the tags. In two dimensions, it's just a line. Everything on one side of the line is red, and everything on the other side is blue. In sentiment analysis, for example, it will be positive and negative. To maximize machine learning, the best hyperplane is the one with the largest distance between each tag. However, as datasets become more complex, it may not be possible to draw a single line to categorize the data into two camps. Using SVM, the more complex the data, the more accurate the predictor will become by increasing the number of dimensions.

After analyzing different approaches to data classification in machine learning, you can choose the one that best suits the task. In the information system being developed, two programs directly need these methods for their functionality: a program for visualization and classification of aerial objects in real-time and a program for classifying objects for radio technical intelligence.

The program for visualizing and classifying aerial objects involves the usual classification based on known data. It is also worth noting that a classification element can have more than two characteristics and all of them will affect the class. After analyzing the known methods of classification, we can conclude that most methods are not suitable for this task. The logistic regression method would not be a bad option, but this method supports linear functions, and in this problem, it is not obvious that there will be a linear dependence. The naive Bayesian classifier method has the same problems in this problem as the previous method. The method of the support vector machine. Of the remaining three methods, the k-nearest neighbours method is the best for this problem, because it is desirable to use it when there are a large number of points. It will be able to use a different number of characteristics and work fast enough.

The program for classification of objects for radio technical intelligence also involves the use of the method of classification of objects based on known data. However, this classifier differs from the classifier described above. This classifier should return the probabilities of the object belonging to the corresponding class. In its general features, it is similar to a classifier containing a program for visualizing and classifying aerial objects, so it can be concluded that the k nearest neighbours method is suitable for this task. However, in this problem, it is necessary to determine the probabilities, and with the help of the method under consideration, it is impossible to do this. The solution to this problem will be a unique classification method based on k nearest neighbours and a naive Bayesian classifier. This approach is based on comparing the distances of objects classified in a multidimensional plane with the axes corresponding to their characteristics. Using this approach, the probability of belonging to the corresponding class based on all distances will be calculated.

4.2. Natural Language Processing Method

Another feature that one of the system programs includes is natural language processing. Natural language processing (NLP) refers to the field of computer science, or rather the field of artificial intelligence, which gives computers the ability to understand text and spoken words almost as well as humans [26]. NLP combines computational linguistics – rule-based modelling of human language – with statistical, machine and deep learning models. Together, these technologies allow computers to process human language in the form of text or voice data and "understand" its full meaning, along with the intent and sentiment of the speaker or writer. NLP manages computer programs that translate text from one language to another, respond to voice commands, and quickly summarize large volumes of text – even in real-time. People are interacting with NLP in the form of GPS voice systems, digital assistants, speech-to-text software, customer service chatbots, and other consumer conveniences. But NLP is also playing a growing role in enterprise solutions that help streamline business operations, improve employee productivity, and simplify mission-critical business processes. Natural language processing includes two approaches: statistical and neural network methods. Much natural language processing research relies heavily on machine learning [27]. The machine learning paradigm involves using statistical inference to automatically learn rules by analyzing large corpora (the plural form of the corpus is a set of documents, possibly with human or computer annotations) of typical real-world examples. However, the main disadvantage of statistical methods is that they require detailed development of features. Thus, the industry has largely abandoned statistical methods and moved to neural networks for machine learning [28]. Popular techniques include using word embedding to capture the semantic properties of words and increase end-to-end learning of a higher-level task (e.g., answering questions) instead of relying on the pipeline of separate intermediate tasks (e.g., part-of-speech tagging and dependency parsing). Natural language processing is a necessary functionality for the program to integrate definitions of potentially unknown terms and abbreviations into informational materials. In this program, it will be rational to use machine learning instead of neural networks. In the program, the methods of processing certain words and phrases for their search should be developed, and not methods for understanding the meaning of sentences or similar tasks, which are better handled by a neural network. The tasks that must be performed relate to a certain area of natural language processing - morphological analysis. It includes topics such as lemmatization, morphological segmentation, part-of-speech marking, and stemming.

Lemmatization is the task of removing only the inflectional endings and returning the basic dictionary form of the word, which is also known as a lemma. This is another method of reducing words to their normalized form. But in this case, the conversion uses a dictionary to display the words in their actual form.

Morphological segmentation – dividing a word into separate morphemes and determining the class of morphemes. The complexity of this task largely depends on the complexity of the morphology (that is, word structure) of the language under consideration. English has a fairly simple morphology, especially inflectional morphology, and so it is often possible to completely ignore this task and simply model all possible word forms (for example, "open, opens, opened, opening") as separate words. However, in highly agglutinated languages such as Turkish or Meitei, an Indian language, this approach is not possible, as each dictionary entry contains thousands of possible word forms.

Part-of-speech marking – part-of-speech identification for each word in the given sentence. Many words, especially common ones, can perform the functions of several parts of speech. For example, "book" can be a noun ("the book on the table") or a verb ("to book a flight"); "set" can be a noun, verb or adjective; and "out" can be any of at least five different parts of speech.

Stemming is the process of reducing declinable (or sometimes derived) words to their basic form (for example, "close" will be the root for "closed", "closing", "closed", "closer", etc.). Root generation produces the same results as lemmatization, but it does so based on rules rather than a dictionary.

All these methods will play a key role in the natural language processing algorithms that will be contained in the information files of the military. These methods will help to connect the dictionary with the text even if the words will be in different cases or phrases.

Each information system program has its unique algorithms. These algorithms need to be analyzed separately. However, there are also non-main algorithms, they organize the work of the program in such a way that it can function correctly. These algorithms are not special, they are formed according to the basic principles of programming and are simply necessary for the system to function. The program for structuring military information involves working out the algorithm for forming a hierarchical menu (Fig. 14a). The algorithm processes page objects that are connected by a one-way connection. The function should process these connections and form a menu object with a hierarchical structure. The algorithm was implemented using recursion, which made it possible to dynamically form sub-items for each menu item.

The program for visualization and classification of aerial objects in real time involves working out the map model formation algorithm (Fig. 14b). The algorithm receives input data from the radar. They contain information about aerial objects. The algorithm processes this data with the help of a classifier that uses the k nearest neighbours algorithm and forms a data structure that will contain a map model with all classified aerial objects in real-time.
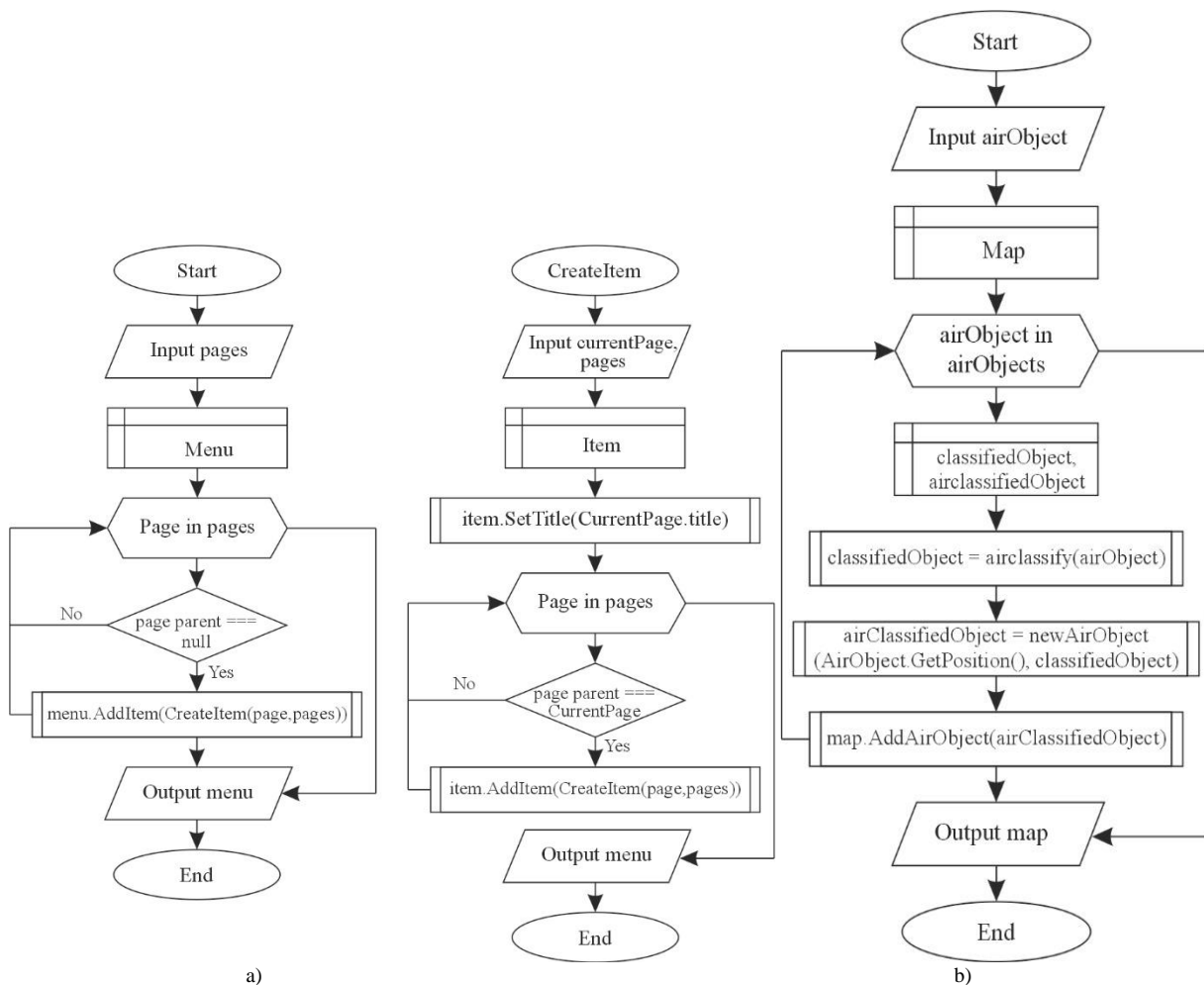

Fig.14. Block diagram of the algorithm for a) structuring of military information and b) visualization and classification of aerial objects in real-time

The program for classifying objects for radio technical intelligence involves the development of a unique classification algorithm (Fig. 15a). The algorithm processes the process formed on the approaches of k-nearest neighbours and naive Bayesian classifier. It calculates error values to determine how far the characteristic is from the target characteristic. Using logarithms and the power function normalizes the distances and makes the permissible range. Also, for the algorithm, each characteristic is the same in terms of weight, so if 3 out of 4 characteristics have a value of 1 and the last one has a value of 0, then the result will be 75%. After all the probability values for the classes are obtained, the algorithm forms an object to return the final data structure with all the information. The program for the integration of definitions of potentially unknown terms and abbreviations into informational materials involves the development of an algorithm for forming the dictionary data structure that connects all words with all definitions (Fig. 15b). The algorithm

receives dictionary data as input in text form to transform this data into a structured object containing a model of the dictionary with which the program will work. The algorithm forms two data structures based on a hash table for keys (dictionary words) and definitions (content associated with a word in the dictionary). A dictionary may have multiple definitions for multiple words. This algorithm defines groups and combines entities to complete several definitions of the same word or phrase. As a result, these objects are combined into a single returned data structure.
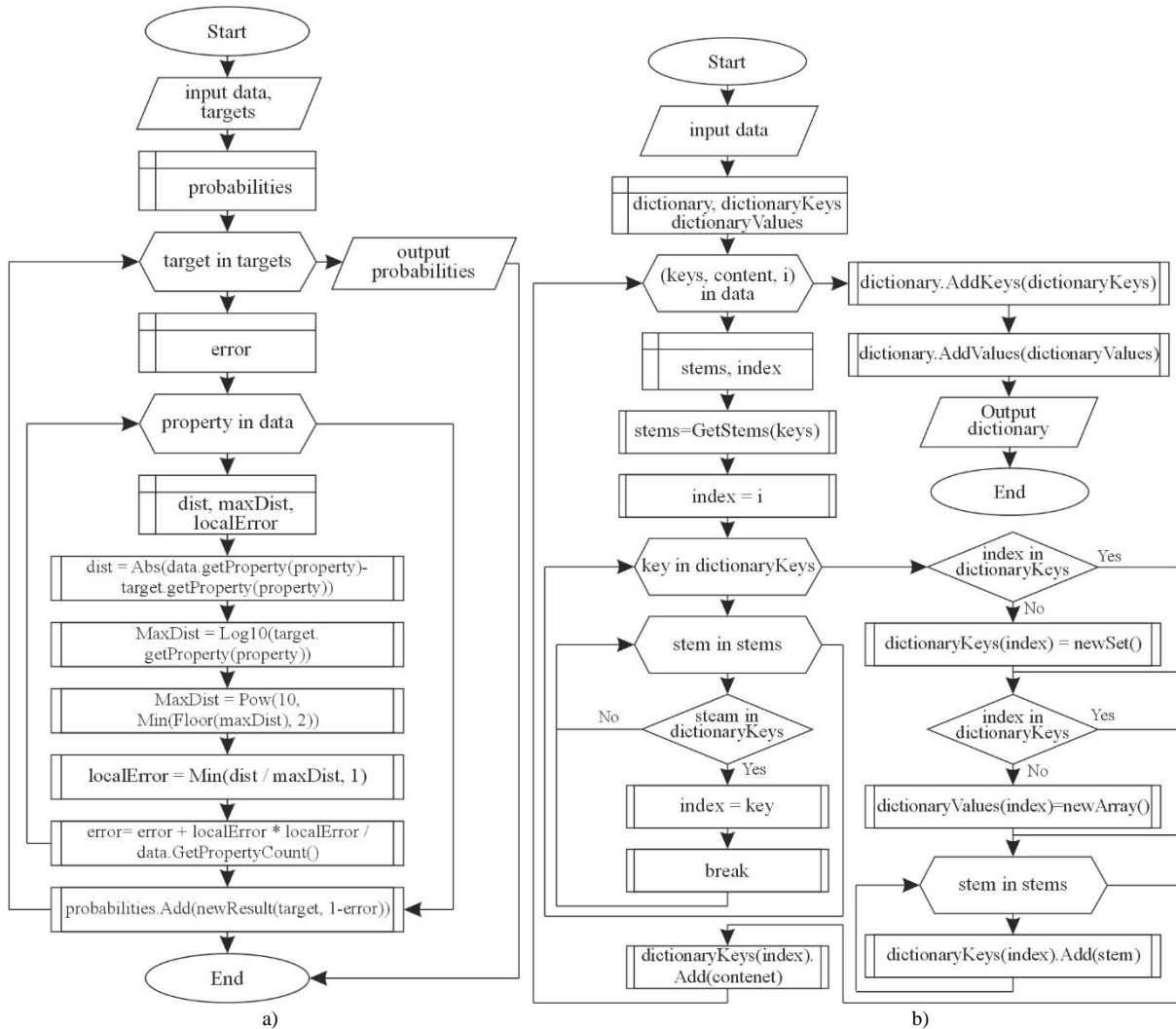
Fig.15. Block diagram of the program algorithm for a) the classification of objects for radio technical intelligence and b) the integration of definitions of potentially unknown terms and abbreviations into informational materials

The developed information system is complex and therefore includes a set of separate independent programs that have nothing in common but are united by a common goal - to process military data to help commanders of tactical units. All these programs are included in one package. Using a bash script, you can easily install these programs on your computer automatically. Because the tools, technologies, and requirements for developing these programs are different, the development process itself and the results are also different. The main part of software development includes two parallel stages. The first stage is the development of the client part. This stage involves the development of an interface through which interaction with the program will take place. Usually, this work is performed by appropriate programmers with a special technology stack [29-31]. The second stage is the development of the server part of the program. At this stage, the main functionality of the program is being developed. The developers responsible for this stage do not care about the client part of the program, they implement the necessary modules and interfaces regardless of the interfaces that will be on the client side [32]. Before developing the project, a workspace was organized. A local XAMPP server was deployed, a database was created (Fig. 16) and connections between tables were organized [33].

After creating the environment, we moved on to the process of writing the software code. First, the layout of the graphic interface was developed (Fig. 17). These templates will be generated for the user to interact with the software visually through the browser.

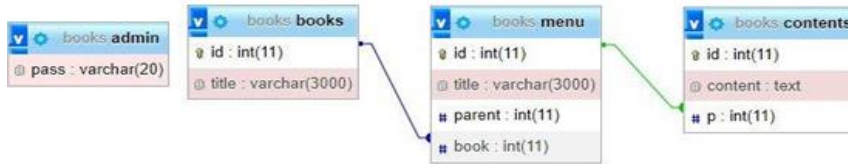Fig.16. Structure of the database

```
388  <body>
389    <div class="sidebar">
390      <ul>
391        <li>
392          <div>
393            <a href="#"><b>Сторінка книги</b></a>
394          </div>
395        </li>
396      </ul>
397    </div>
398
399    <main>
400      Конент сторінки
401    </main>
402
403    <script type="text/javascript">
404      function setCookie(cname, cvalue, exdays) {
405        var d = new Date();
406        d.setTime(d.getTime() + (exdays*24*60*60*1000));
407        var expires = "expires="+ d.toUTCString();
408        document.cookie = cname + "=" + cvalue + ";" + expires + ";path=/";
409      }
410
411      function getCookie(cname) {
412        var name = cname + "=";
413        var decodedCookie = decodeURIComponent(document.cookie);
414        var ca = decodedCookie.split(';');
415        for(var i = 0; i <ca.length; i++) {
```

Fig.17. Fragment of page layout code

```
388  <body>
389    <div class="sidebar">
390      <div class="controls">
391        <a href="add" style="background: #13ce66;">Додати</a>
392        <a href="delete" style="background: #f00;">Видалити</a>
393        <a href="rename" style="background: #aaa;">Змінити</a>
394      </div>
395      <ul>...</ul>
396    </div>
397
398    <main>
399      <form method="POST" action="#">
400        <textarea name="editor" id="editor" rows="10" cols="80"></textarea>
401        <div class="submit">
402          <button type="submit">Зберегти</button>
403        </div>
404      </form>
405    </main>
```

Fig.18. Fragment of the page layout code for the administrator

```
1   <?php
2   $admin = false;
3
4   session_start();
5
6   $conn = new mysqli("localhost", "*********", "*********", "books");
7   $conn->set_charset("utf8");
8   if ($conn->connect_error) {
9     die("Connection failed: " . $conn->connect_error);
10  }
11
12  if (isset($_GET["exit"])) {
13    session_destroy();
14    header("location: index.php");
15  } else if (isset($_GET["admin"]) && !isset($_SESSION["admin"])) {
16    if (strlen($_GET["admin"]) > 0) {
```

Fig.19. A fragment of the database connection code

```
489  <body>
490    <div class="sidebar">
491      <?php
492        if (isset($_SESSION["admin"]) && $_SESSION["admin"] == 1) {
493          ?>
494      <div class="controls">
495        <a href="add" style="background: #13ce66;">Додати</a>
496        <a href="delete" style="background: #f00;">Видалити</a>
497        <a href="rename" style="background: #aaa;">Змінити</a>
498      </div>
499          <?php
500        }
501      ?>
502      <ul <?php if(isset($_SESSION["admin"]) && $_SESSION["admin"] == 1) {echo
            "id=\"bookList\"";} ?>>
503    <?php
504      for ($i=0; $i<count($mainData[0]); $i++) {
505        ?>
506        <li data-info="<?php echo ($i+1); ?>" <?php if ($page ==
             $mainData[0][$i]["id"]) {echo "class=\"active\"";} ?>>
507          <div>
508            <?php if (isset($mainData[$mainData[0][$i]["id"]])) { ?>
509              <button class="sh"></button>
510              <div class="sh-l"></div>
511            <?php } ?>
```
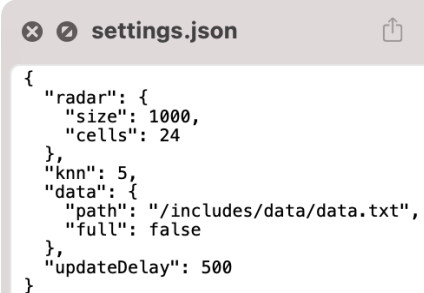
Fig.20. A fragment of the program code

To create a convenient user interface for administrators, an ordinary user page was taken and additional components were built on top of it, which only the administrator will see (Fig. 18). Thus, the graphical interface will be clear and easy to use. After developing the layout of the graphical representation of the program, the logical part of the program was developed. Using PHP, the program was connected to the database (Fig. 19) and the generation of the corresponding part of the graphic interface was worked out (Fig. 20).

```
settings.json

{
  "radar": {
    "size": 1000,
    "cells": 24
  },
  "knn": 5,
  "data": {
    "path": "/includes/data/data.txt",
    "full": false
  },
  "updateDelay": 500
}
```

Fig.21. Configuration file

```
1   <!DOCTYPE html>
2   <html lang="en">
3   <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <title>Radar</title>
7     <link rel="stylesheet" href="includes/css/common.css">
8   </head>
9   <body class="loading">
10    <div id="container">
11      <div></div>
12    </div>
13
14    <div class="legend">
15      <div class="class">*****</div>
16      <div class="class red">*******</div>
17    </div>
18
19    <script src="renderer.js"></script>
20  </body>
21  </html>
```
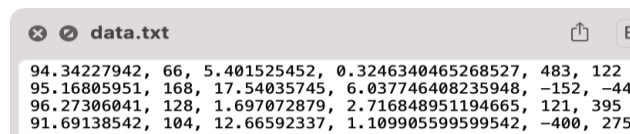
Fig.22. Graphical interface code

```
9   const createWindow = () => {
10    mainWindow = new BrowserWindow({
11      width: 1000,
12      height: 600,
13      minHeight: 400,
14      minWidth: 400,
15      icon: `${__dirname}/includes/img/icon.ico`,
16      webPreferences: {
17        preload: `${__dirname}/preload.js`,
18        nodeIntegration: true,
19        contextIsolation: false,
20        enableRemoteModule: true,
21      },
22    });
23
24    mainWindow.loadURL(`file://${__dirname}/index.html`);
25    mainWindow.setMenuBarVisibility(false);
26
27    mainWindow.on('closed', () => {
28      mainWindow = null;
29    });
30  };
31
32  app.on('ready', createWindow);
33
34  app.on('window-all-closed', () => {
35    if (process.platform !== 'darwin') {
36      app.quit();
37    }
38  });
```

Fig.23. A fragment of program code

```
⊗ ⊘  data.txt                                          ⬆  B
94.34227942, 66, 5.401525452, 0.3246340465268527, 483, 122
95.16805951, 168, 17.54035745, 6.037746408235948, −152, −44
96.27306041, 128, 1.697072879, 2.716848951194665, 121, 395
91.69138542, 104, 12.66592337, 1.109905599599542, −400, 275
```

Fig.24. File for communication between radar and program

At this stage, the program was ready, since XAMPP can run programs written in PHP and return a response using HTTP requests. Before developing the project, a workspace was organized. NodeJS was installed and necessary packages such as ElectronJS, BrainJS and others were imported. A configuration file was also created for convenient setup (Fig. 21). Thus, it became possible to manage the capabilities of the program. After organizing the environment, the graphical interface of the program was created (Fig. 22). It was made using web technologies because ElectronJS makes it possible to run programs on a PC.

The next step was to develop the business logic of the application. Using the framework, the general work of the program was organized (Fig. 23). Next, a code was developed that forms a map for displaying classified aerial objects. To organize work in real-time, it was necessary to connect the radar with the program [34]. Since many radars cannot operate over a permanent open channel, a middleman approach was used. The program should only read data from a specified file at a certain time interval, and the task of the radar or the radar adapter program is to edit this file (Fig. 24). The file must contain all the data provided by the radar. Thus, the program used one interface for working with different types of radars.

```
81   fs.readFile(`${__dirname}/includes/data/train/train.csv`, 'utf8', (err, data) =>
       {
82     if (err) {
83       console.error(err);
84       return;
85     }
86
87     const rows = data.split('\n');
88     const input = [];
89     for (const row of rows) {
90       input.push(row.replace(/\s/g, '').split(';').map(parseFloat));
91     }
92
93     const inputData = [];
94     const outputData = [];
95     for (const row of input) {
96       if (row.length < 4) { continue; }
97       inputData.push(row.slice(0, -1));
98       outputData.push(row.slice(-1)[0]);
99     }
100
101    knn = new KNN(inputData, outputData, { k: window.knn });
102
103    const jsData = knn.toJSON();
104    fs.writeFile('includes/data/knn/knn.json', JSON.stringify(jsData), 'utf8', ()
         => {
105      console.log('done');
106    });
107  });
```

Fig.25. Code fragment for creating a classifier model

The next step was to set up the classification. To classify objects, you need to have a working classifier, and for it to work, you need to first form it. Accordingly, the code for creating a classifier (Fig. 25) was written so that it formed a data structure (Fig. 26) which will be used later by the program for classifying aerial objects. The last step was to create an executable file, using the operations provided by the CLI of the ElectronJS framework, to run the program on the Windows operating system. Before the development of the project, a workspace was organized for the development of a program for the classification of objects for radio-technical intelligence. A local XAMPP server has been deployed to use the application in a browser over a local network. Next, the layout of the graphic interface was developed (Fig. 27).

Fig.26. Classifier model file

```
23    <div style="display: inline-block; padding: 5px;">
24      <div style="font-size: 14px;">*********:</div>
25      <input data-input placeholder="0" value="0" style="padding: 10px;border: 2px solid
          #ddd;border-radius: 5px;font-size: 16px;display: block;width: 100%;margin-top: 5px;" type="text">
26    </div>
27  </div>
28
29  <hr>
30
31  <div style="font-size: 20px; line-height: 1.5;">*******: <b>0%</b></div>
32  <div style="font-size: 20px; line-height: 1.5;">********: <b>0%</b></div>
33  <div style="font-size: 20px; line-height: 1.5;">*********: <b>0%</b></div>
34
35  <script>
36    (function(){
37      const fade = 2;
38
39      const inputs = $("*[data-input]");
40      const entities = $("*[data-entity]");
41
42      for(let i = 0; i < inputs.length; i++) {
43        $(inputs[i]).on("keyup", function(){
44          input();
45        });
46      }
```

Fig.27. Layout of the graphic interface

```
35    <script>
36      (function(){
37        const fade = 2;
38
39        const inputs = $("*[data-input]");
40        const entities = $("*[data-entity]");
41
42        for(let i = 0; i < inputs.length; i++) {
43          $(inputs[i]).on("keyup", function(){
44            input();
45          });
46        }
47
48        function input() {
49          const data = [];
50          for(let i = 0; i < inputs.length; i++) {
51            let v = parseFloat(inputs[i].value) || 0;
52            data.push(v);
53            if(inputs[i].value[inputs[i].value.length - 1] == ".") v += ".";
54            inputs[i].value = v;
55          }
56
57          update(data);
58        }
```

Fig.28. A fragment of the program code

For the operation of the program, the code was written in the JavaScript language (Fig. 28) and for convenient and quick extension of the program, attributes were used that provide information about the class (Fig. 29). Thus, it is only necessary to add new objects with attributes to obtain the probability of the input object belonging to these new classes.

```
40  <b data-entity="[[0.2,18],[0.2],[2],[10]]">0%</b>
41  <b data-entity="[[1,18],[3],[5],[20]]">0%</b>
42  <b data-entity="[[0.03,1.7],[2],[2],[1000]]">0%</b>
43
44  <script>
45    (function(){
46      const fade = 2;
47
48      const inputs = $("*[data-input]");
49      const entities = $("*[data-entity]");
50
51      for(let i = 0; i < inputs.length; i++) {
52        $(inputs[i]).on("keyup", function(){
53          input();
54        });
55      }
```

Fig.29. Code fragment of classification objects

Since the application was written as a static resource on the server, XAMPP can run the application using a simple HTTP request in the browser.

Before the development of the program, a workspace is organized for the integration of definitions of potentially unknown terms and abbreviations into the informational materials. Python was installed and necessary packages such as Tkinter, Fitz and others were imported. Also, before developing the code, a prototype was developed in the Figma design tool (Fig. 30). In this way, a convenient and beautiful graphical interface of the program was selected [35].
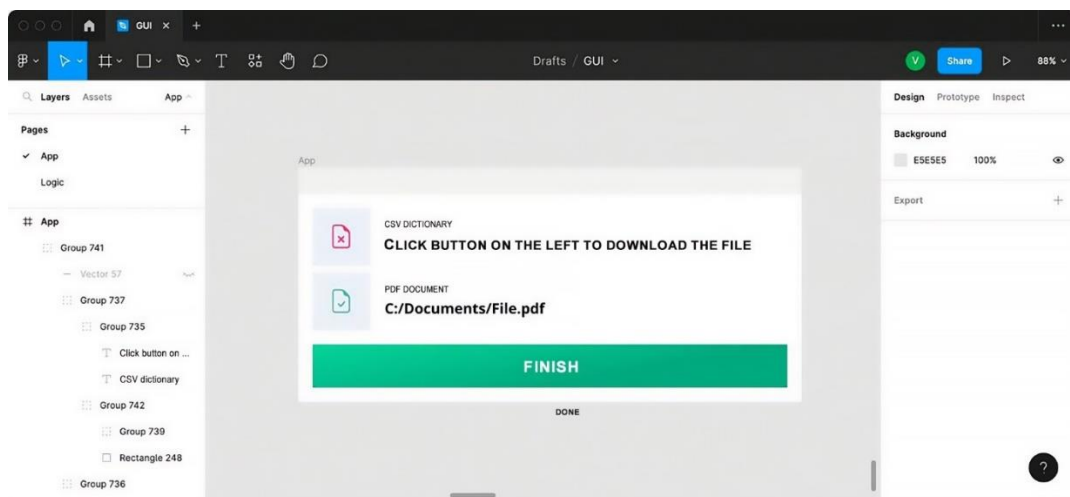

Fig.30. Graphical interface design

After creating the design, the layout of the graphic interface was developed using the Tkinter library (Fig. 31). The code of the graphical interface of the program is presented in Appendix A. Thus, it became possible to open the program and use it using the graphical interface. The next step was the development of the main program code (Fig. 32). Thanks to this code, the program has all the functionality that was described in the requirements. The last step was to create an executable file using Python capabilities to run the program on the Windows operating system. After developing an information system, one of the important next steps is software testing. This is especially important in the military field because even small flaws can do a lot of damage. Software testing is important because if there are any bugs in the software, they can be detected at an early stage and resolved before the software product is shipped. A properly tested software product ensures reliability, security and high performance, which further leads to time savings, cost efficiency and customer satisfaction. Implementation of the project is also a mandatory stage. For users to be able to use software products, they need to be deployed on the Internet or provide information and tools for their installation. Deployment is the mechanism by which applications, modules, updates, and patches are delivered from developers to users. The methods developers use to build, test, and deploy new code will affect how quickly the product can respond to changes in customer preferences or requirements, as well as the quality of each change.

```
304  csvTitleLabel = Label(root, text="CSV СЛОВНИК", bg='#ffffff', font=titleFont)
305  csvPathLabel = Label(root, textvariable=csvPath, bg='#ffffff', font=pathFont)
306
307  pdfTitleLabel = Label(root, text="PDF ДОКУМЕНТ", bg='#ffffff', font=titleFont)
308  pdfPathLabel = Label(root, textvariable=pdfPath, bg='#ffffff', font=pathFont)
309
310  csvButton.place(x=20, y=20)
311  pdfButton.place(x=20, y=110)
312  saveButton.place(x=20, y=210)
313
314  csvTitleLabel.place(x=120, y=34)
315  csvPathLabel.place(x=120, y=56)
316
317  pdfTitleLabel.place(x=120, y=124)
318  pdfPathLabel.place(x=120, y=146)
319
320  root.mainloop()
```

Fig.31. Graphical interface code fragment

```
 9  def run(pathes):
10      dictionary_keys = dict()
11      dictionary_values = dict()
12
13      with open(pathes["csv"], 'r') as file:
14          reader = csv.reader(file, delimiter = ";")
15
16          i = -1
17          for (keys, input) in reader:
18              i += 1
19              stems = []
20              keys = keys.strip().split(",")
21
22              for key in keys:
23                  tokens = re.findall(r'\[?\w+\]?', key.strip())
24                  if ord(tokens[0][1]) < 127:
25                      for j in range(len(tokens)):
26                          tokens[j] = USStemmer.stem(tokens[j])
27                  else:
28                      for j in range(len(tokens)):
29                          tokens[j] = UAStemmer.stem(tokens[j])
30                  stems.append(' '.join(tokens))
31
32              def get_index():
33                  for key in dictionary_keys.keys():
34                      for stem in stems:
35                          if stem in dictionary_keys[key]:
36                              return key
37                  return i
38
39              index = get_index()
```

Fig.32. A fragment of the program code

To carry out the project testing process, two stages were developed: verification and validation. First, verification was performed to confirm the compliance of the project with the tasks. All requirements related to the project were worked out and verified. As a result, a numerical percentage value of the evaluation of the conformity of the result to the requirements of the project was obtained. Since the information system includes several autonomous programs, they were all analyzed separately (Table 2) and based on these values, the final value for the information system developed in this work was calculated.

Table 2. Compliance of the software with the requirements that were set

The name of the information system component is the program for	Compliance ratings
structuring of military information	93%
visualization and classification of aerial objects in real-time	90%
classification of objects for radio technical intelligence	95%
integration of definitions of potentially unknown terms and abbreviations into informational materials	91%

After receiving the results, you can pay attention to the fact that all the compliance ratings are greater than or equal to 90%, which means that the software tools are excellent in terms of their compliance with the requirements described above. Although each program has high marks, it does not reach 100%. This means that some requirements were not fully met. For the program for structuring military information, the goal was to quickly find the necessary information due to the hierarchical structure of the pages. This requirement was met, but some points and details of page tree visualization were missed. These are details such as highlighting the path of the page in its active state or when hovering over it.

Real-time data processing was the goal of the program for the visualization and classification of aerial objects in real-time. This requirement was met, but some things were simplified. Ideally, as soon as the radar receives data, the program receives this data at the same time, but because there are different types of radars, we had to make interval checks for data updates. Thus, there is a very small time range in which the data may not be relevant.

For the object classification program for radio technical intelligence, the goal was to find the probabilities of the input object belonging to a certain class. This requirement has been met, but with specific data, the probabilities may not be as accurate as they could be. The classifier algorithm looks for a range in which it compares characteristics. This range is calculated from the value of the target characteristic, and with specific data, the ranges can be either too small or too large, which, although not much, will affect the value of the probabilities.

For a program to integrate definitions of potentially unknown terms and abbreviations into information materials, the goal was to process information files to integrate a certain dictionary into them. This requirement was met, however, only for the PDF format. Although this type of file is most common in the field of information materials, there are still several extensions that can be used. It is not possible to use these extensions in this version of the program, however, the system architecture allows you to easily add new implementations for different types of files, so this indicator will increase in the future.

After summarizing the evaluations of all the software tools included in the complex information system for processing military data to help commanders of tactical units, a compliance value of 94% was obtained, and therefore the system successfully passed the verification stage.

The next step was project validation. Validation confirms the ability of the tools developed within the scope of the project to implement the process of solving the specified task under specified conditions while obtaining the appropriate, relevant and correct result by confirming compliance with the functional requirements, as well as by performing a control solution of the project task. Since the BDD (Behavior Driven Development) approach was used during the development of the project, User Stories were used for testing.

Behavior-driven development (BDD) is a flexible software development methodology, according to which a program is documented and developed according to the behavior expected by the user when interacting with it [36]. By encouraging developers to focus only on the intended behaviour of the application, BDD helps avoid bloat, excessive code, unnecessary features, or insufficient focus. This methodology combines, complements, and improves the practices used in test-driven development (TDD) and acceptance testing. A typical project that uses behaviour-driven development begins with a conversation between developers, managers, and the customer to form a big picture of how the product should work. Expectations for product behaviour are then set as goals for developers, and after all behavioural checks have been passed, the product is compliant and ready to be delivered to the customer. BDD offers the opportunity to expand the pool of input and feedback to include business stakeholders and end users who may not have sufficient knowledge of software development. With this enhanced feedback loop, development teams can more easily use BDD in continuous integration and continuous delivery environments. As for the mathematical part of algorithm testing, test data (Fig. 33-34) were used to check the operation of all algorithms that involve working with data. After completing these stages of testing, the information system was additionally tested by testers. The system was tested by people in real conditions to simulate the behaviour of real potential users.
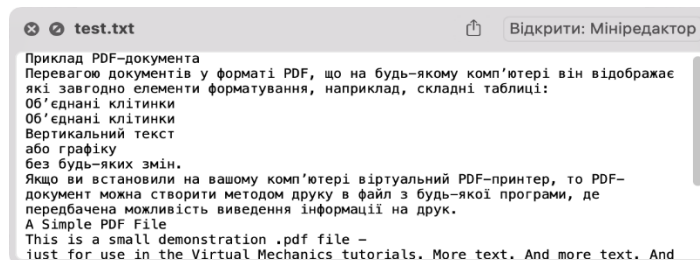

Fig.33. One of the test data files

Fig.34. Test data for visualization and classification

Since the information system involves work in a local network to which the developers do not have access, a package with all programs and installers will be delivered to the end user. It is important to take into account that you need to use the Windows operating system to work with all system programs. To install an information system for processing military data to help commanders of tactical units, the user only needs to run a bash script that will install all programs included in the information system itself. Thanks to the graphical interface of all programs included in the information system being developed, users are unlikely to have any problems using the programs. The program for structuring military information has several pages that provide all the necessary functionality. After opening the web application through the browser, the user gets to the book selection page (Fig. 35). After selecting a book, the user goes to view the book with a visualization of its structure in the left part of the page (Fig. 36). Using the menu on the left, the user can move from page to page. To manage the information contained in the application, the user must be authorized to use the application as an administrator. After that, all pages accessible to a simple user will have additional controls (Fig. 37) with which you can add, delete, and edit information.
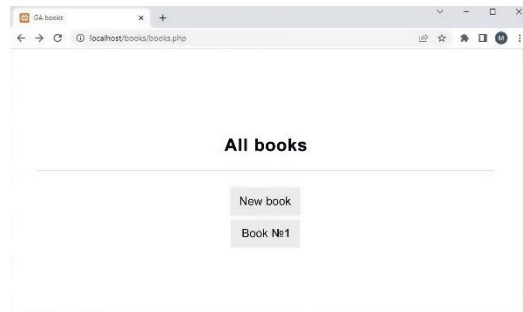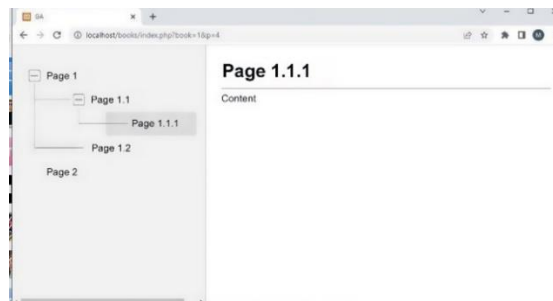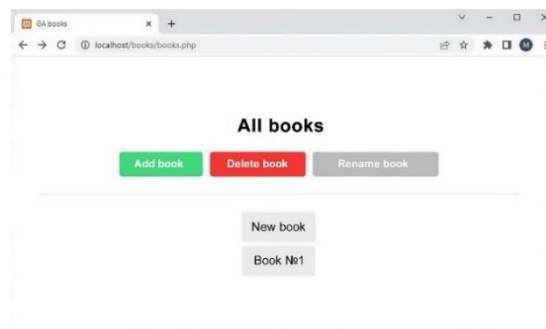

Fig.35. Book selection

Fig.36. Viewing the book

Fig.37. Administrator controls

The program for visualization and classification of aerial objects in real time has a graphical interface consisting of one window (Fig. 38). This window shows air objects already classified and placed in the correct places, the data of which was received from the radar. This window is updated periodically. This period and other settings can be set in the configuration file described in the previous section. Information about aerial objects, their name and rendering colours is also displayed at the bottom left.

The program for the classification of objects for radio technical intelligence also has one window through which interaction with the program takes place. After opening the program in the browser, the user gets to the program page (Fig. 39) where he can enter parameters for the classifier. The classifier reacts to the change of each parameter, which means that after each change of the characteristic, according to the current data, classification will be carried out and the probability of belonging of the object with the characteristics entered in the form to the existing classes will be derived. The program for integrating definitions of potentially unknown terms and abbreviations into information materials, like

the two previous programs, has one window for user interaction with the program (Fig. 40). However, unlike other programs, this program involves first creating a dictionary (Fig. 41) in any convenient program that allows you to export data with the CSV type. This file, as well as the information file with the PDF type, will be loaded into the program to combine them. As a result, a copy of the information file with additional data appears on the user's computer, which is displayed in the browser as notes (Fig. 42).
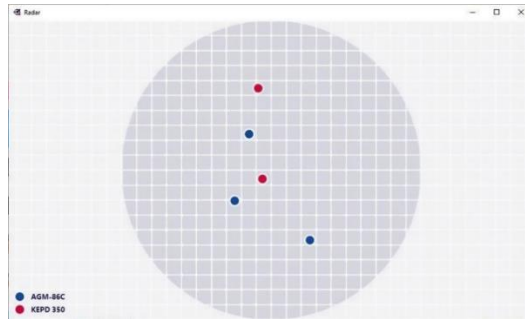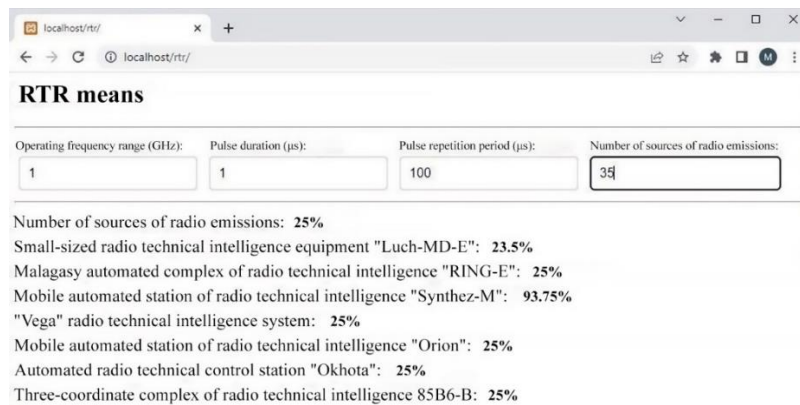

Fig.38. Graphical program interface

Fig.39. Graphical interface of the program for classification of objects for radio technical intelligence

Fig.40. Graphical interface

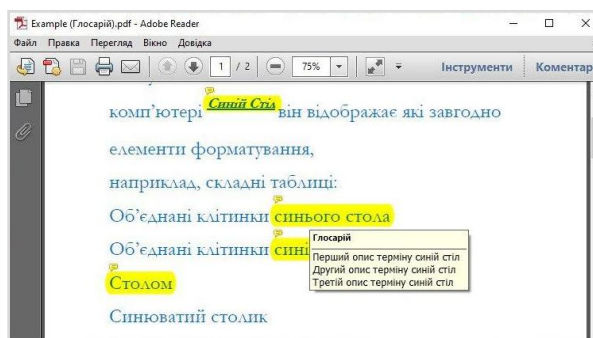Fig.41. Dictionary - data on words and phrases

Fig.42. Viewing the information file generated by the program

In this way, the user can view the document without unnecessary information, and when this information is needed, he can find it by hovering the cursor over the relevant word.

5. Conclusions

In the course of the work, the relevance of the chosen topic was investigated and revealed. The process and various aspects of the work of the commander of tactical units were analyzed and potential problems that he may face during his work were identified. To solve the problems, a solution was presented in the form of an information system for processing military data to help commanders of tactical units. During the analysis of the current state and prospects in the field of research, the details of the process of command and control carried out by the commander of tactical units were elaborated in more detail. The closest analogues that existed at the time of the project development were also analyzed. Their advantages and disadvantages were determined, and based on these data, the final vision of the information system was formed. To correctly create the architecture and highlight the functional tasks of the system being developed, a system analysis was performed and a conceptual model was built. Before that, all the requirements for the information system were defined and documented. Using this data in combination with information about the purpose, the problems the product should solve and its overall vision, all the necessary diagrams were built using UML notation. These diagrams show the relationships between objects and the overall architecture of the system. The architecture of the system was built in such a way that the component systems and the system as a whole could be easily expanded without spending a lot of time and resources on it. Diagrams summaries all information about the processes and objects of the information system in a visual representation. Appropriate means of project development are selected, described and substantiated for convenient development. The algorithms programmed in the information system are described and visualized on the diagrams. All the main methods used to achieve certain tasks in the project were defined and described. In particular, for the effective operation of some parts of the system, a new method of classification based on known ones was presented. Its idea and work algorithm are presented. During and after the development of the project, testing of the information system was carried out to confirm the fulfilment of all requirements. The degree of implementation of each component of the information system was evaluated and this information was analyzed. At the testing stage, the final product was verified and validated, which formulated a conclusion about the work result. At the end of the work, information on the implementation of the project was provided. The process of operation of the components of the system from the side of the end-user and the process of deployment by the end user of the information system for processing military data to help the commanders of tactical units were described.

Conflict of Interest

The authors declare no competing interests.

References

[1] Gurney, Jennifer M., et al. Tactical combat casualty care training, knowledge, and utilization in the US Army. Military Medicine, 2020, 185.Supplement_1: 500-507.
[2] Halyna Antonyuk, Lilia Chernysh, Diachronical Analysis of Lexicographic Sources of Military Terminology, CEUR Workshop Proceedings, Vol-3171, 2022, pp. 664-676
[3] Barbar, Ana Elisa. Challenges for Ethical Humanitarian Health Responses in Contemporary Conflict Settings. Dædalus, 2023, 152.2: 53-62.
[4] Korolov, Volodymyr, et al. Information-Reference System Creation Prerequisites for the Ground Forces Identification on the Battlefield According to NATO Standards. In: COLINS. 2021. p. 1152-1172.
[5] O. Klymovych, V. Hrabchak, O. Lavrut, T. Lavrut, V. Lytvyn, V. Vysotska, The Diagnostics Methods for Modern Communication Tools in the Armed Forces of Ukraine Based on Neural Network Approach, volume Vol-2631 of CEUR Workshop Proceedings, 2020, pp. 198-208.
[6] O. Pashchetnyk, V. Lytvyn, V. Zhyvchuk, L. Polishchuk, V. Vysotska, Z. Rybchak, Y. Pukach, The ontological decision support system composition and structure determination for commanders of land forces formations and units in Ukrainian armed forces, volume Vol-2870 of CEUR Workshop Proceedings, 2021, pp. 1077-1086.
[7] Y. Zdorenko, O. Lavrut, T. Lavrut, V. Lytvyn, Y. Burov, V. Vysotska, Route Selection Method in Military Information and Telecommunication Networks Based on ANFIS, volume Vol-2917 of CEUR Workshop Proceedings, 2021, pp. 514-524.
[8] O. Sharko, M. Sharko, O. Liubchuk, G. Krapivina, O. Gonchar, N. Advokatova, O. Zaitseva, Building an Intelligent System for Managing Emigration Labor Resources in Conditions of Uncertainty of Military Actions Based on Markov Chains, CEUR Workshop Proceedings, Vol-3403, 2023, 375-386.
[9] Nastakalov, A. R. (2021). The assess the fault-tolerance, efficiency and performance of military field local network topology. Advanced Information Systems, 5(3), 81–85. https://doi.org/10.20998/2522-9052.2021.3.11
[10] Talibov A. & Guliyev B. (2021). A method for assessing the military-economic indicators with the purpose of locating a logistics center for redeploying troops. Advanced Information Systems, 5(2), 152–158. https://doi.org/10.20998/2522-9052.2021.2.23
[11] Bidyuk, P., Gozhvi, A., Kalinina, I., Modeling military conflicts using Bayesian networks. In: 2018 IEEE 1st International Conference on System Analysis and Intelligent Computing, SAIC 2018 - Proceedings, 2018, 8516861
[12] Butko, I. (2021). The use of geospatial information by public authorities to support the decision making of management. Advanced Information Systems, 5(1), 39–44. https://doi.org/10.20998/2522-9052.2021.1.05

[13] Tverytnykova, E., Demidova, Y., & Drozdova, T. (2021). Management system of occupational safety at Ukrainian enterprises: international and European dimension. Advanced Information Systems, 5(1), 45–53. https://doi.org/10.20998/2522-9052.2021.1.06

[14] Chystov, V., Zakharchenko, I., Pavlenko, V., & Pavlenko, M. (2021). A method for choosing a strategy for the behavior of a cellular automaton when solving the problem of finding targets by a group of moving objects. Advanced Information Systems, 5(4), 42–48. https://doi.org/10.20998/2522-9052.2021.4.06

[15] F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal. Pattern-Oriented Software Architecture. 2002.

[16] L. Bass, P. Clements, R. Kazman. Software Architecture in Practice. Addison-Wesley, 2003.

[17] Evergreen. UML. URL: https://evergreens.com.ua/ua/articles/uml-diagrams.html.

[18] Jacobsen, Ivar, Magnus Christerson, Patrik Jonsson, Gunnar Overgaard. Object Oriented Software Engineering. 1992.

[19] Nina.az. Hierarchical data model. 2021. URL: https://www.wiki.uk-ua.nina.az/Ієрархічна_модель_даних.html

[20] Evergreen. Classical machine learning: tasks of classification, generalization, clustering of data. 2019. URL: https://evergreens.com.ua/ua/articles/classical-machine-learning.html.

[21] Gazette. Logistic regression: model and methods. 2021. URL: https://gazette.com.ua/edu/regresiya-logistichna-model-i-metodi.html.

[22] Nina.az. Naive Bayesian classifier. 2024. URL: https://www.wiki.uk-ua.nina.az/Наївний_баєсів_класифікатор.html.

[23] Nina.az. The method of k-nearest neighbors. 2024. URL: https://www.wiki.uk-ua.nina.az/Метод_найближчих_k-сусідів.html.

[24] Geeksforgeeks. Decision Tree. 2022. URL: https://www.geeksforgeeks.org/decision-tree/.

[25] Nina.az. Method of support vectors. 2024. URL: https://www.wiki.uk-ua.nina.az/Опорно_векторна_машина.html.

[26] Gyansetu. Intro to NLP in Machine Learning. 2024. URL: https://gyansetu.in/blogs/what-is-natural-language-processing/.

[27] Trevor Hastie, Robert Tibshirani and Jerome H. Friedman. The Elements of Statistical Learning, Springer, 2001.

[28] Stuart Russell & Peter Norvig. Artificial Intelligence – A Modern Approach. Prentice Hall. 2002.

[29] Visual Studio Code. Why Visual Studio Code? June, 2021. URL: https://code.visualstudio.com/docs/editor/whyvscode.

[30] Simplilearn. Node.js vs. Java: Differences, Applications, and Why You Should Learn Them. 2021. URL: https://www.simplilearn.com/node-js-vs-java-article/.

[31] Coursera. What Does a Front-End Developer Do? 2022. URL: https://www.coursera.org/articles/front-end-developer/.

[32] Coursera. What Does a Back-End Developer Do? 2022. URL: https://www.coursera.org/articles/back-end-developer/.

[33] Pelekh, I., Berko, A., Andrunyk, V., Chyrun, L., Dyyak, I.Design of a system for dynamic integration of weakly structured data based on mash-up technology, Proceedings of the 2020 IEEE 3rd International Conference on Data Stream Mining and Processing, DSMP 2020, 2020, pp. 420–425. DOI: 10.1109/DSMP47368.2020.9204160

[34] SearchUnifiedCommunications. What is a real-time application (RTA)? Definition and Examples. May 2022. URL: https://www.techtarget.com/searchunifiedcommunications/definition/real-time-application-RTA/.

[35] Smashing magazine. What Is User Experience Design? 2010. URL: https://www.smashingmagazine.com/2010/10/what-is-user-experience-design-overview-tools-and-resources/

[36] Carlos Solis, Xiaofeng Wang. A Study of the Characteristics of Behavior Driven Development. 2011.

Authors' Profiles
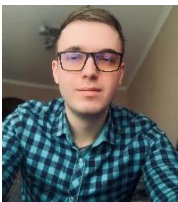
Vitaliy Danylyk, PhD student at the Department of Information Systems and Network at Lviv Polytechnic National University. His research interests span software engineering, data science, system analysis, artificial intelligence and computer science.

Victoria Vysotska, Associate Professor at Information Systems and Networks Department of Lviv Polytechnic National University. She defended her Doctoral degree in Technical Science, specialty in «structural, applied and mathematical linguistics» in 2023 on the topic "Analysis and synthesis of computational linguistic systems for processing Ukrainian textual content" Also, she received a PhD degree in Information Technologies from Lviv Polytechnic National University in 2014. She has currently published more than 490 publications. Her main research interests are focused on the identification of disinformation/fakes/propaganda, detection of sources of disinformation and inauthentic behaviour (bots) of coordinated groups based on artificial intelligence, NLP, computer linguistics, data science, system analysis, information technologies, machine learning cyber security.

Vasyl Andrunyk, Associate Professor at Information Systems and Networks Department of Lviv Polytechnic National University. He received a PhD degree in Information Technologies from Lviv Polytechnic National University in 2021. He has currently published more than 80 publications. His main research interests are focused on Data visualization, Computer systems for processing textual and graphic information, Computer graphics and Virtual reality technologies, Information technologies to support education of persons with autism, and Inclusive education.

Dmytro Uhryn, Associate Professor at the Chernivtsi National University. He received his Bachelor's and Specialist's degrees in Applied Mathematics from Chernivtsi National University in 2002 and 2003, respectively. PhD in Mathematical and Software Engineering of Computers and Systems (2010). Doctor of Technical Sciences in Information Technology (2021). Research interests: data mining, decision support information technologies, swarm intelligence systems, industry-specific geographic information systems.

Yuriy Ushenko, PhD in Optics and Laser Physics (2006). D.Sc. in Optics and Laser Physics, Taras Shevchenko National University of Kyiv (2015). Current position – Professor, Head of Computer Science Department, Chernivtsi National University, Ukraine. Research Interests: Data Mining and Analysis, Computer Vision and Pattern Recognition, Optics & Photonics, Biophysics.