UDC 004.891.2:614.446

K.O. BAZILEVYCH
M.Ye. Zhukovsky National Aerospace University "Kharkiv Aviation Institute", Kharkiv,
Ukraine, e-mail: *ksenia.bazilevich@gmail.com*.

D.I. CHUMACHENKO
M.Ye. Zhukovsky National Aerospace University "Kharkiv Aviation Institute", Kharkiv,
Ukraine, e-mail: *dichumachenko@gmail.com*.

L.F. HULIANYTSKYI
V.M. Glushkov Institute of Cybernetics of NAS of Ukraine, Kyiv, Ukraine,
e-mail: *hulianytskyiLF@nas.gov.ua*.

I.S. MENIAILOV
M.Ye. Zhukovsky National Aerospace University "Kharkiv Aviation Institute", Kharkiv,
Ukraine, e-mail: *evgenii.menyailov@gmail.com*.

S.V. YAKOVLEV
M.Ye. Zhukovsky National Aerospace University "Kharkiv Aviation Institute", Kharkiv,
Ukraine; Lodz University of Technology, Lodz, Poland,e-mail: *svsyak7@gmail.com*.

INTELLIGENT DECISION SUPPORT SYSTEM FOR EPIDEMIOLOGICAL DIAGNOSTICS. II. INFORMATION TECHNOLOGY DEVELOPMENT[1,2]

Abstract. The article projects the components of the intelligent decision support system for epidemiological diagnostics and investigates their interaction with the user. The system includes a bank of models and machine learning methods, a bank of population dynamics models, visualization and reporting tools, and management decision-making unit. The concept of information technology to ensure biosafety of the population is provided. A model of specified information technology use cases is developed and a sequence diagram is constructed. A model of information technology components and ways of their deployment on a server are proposed.

Keywords: decision support system, information technology, epidemiological diagnostics, machine learning, population dynamics.

INTRODUCTION

This paper continues the study [1] devoted to the development of an intelligent decision support system for epidemiological diagnostics. Based on the analysis of the peculiarities of decision-making in the field of epidemiology and public health, the architecture of such an information system was proposed, taking into account the formalization of its main characteristics. In general, this research is based on the concept of developing a decision support system for the control of epidemic morbidity, proposed in [2].

The COVID-19 pandemic dictates the need to develop an integrated intelligent decision support system for epidemiological diagnostics. The introduction and use of such an information system will make it possible to make timely and scientifically sound decisions on the use of control measures that restrain the development of epidemics. The proposed information system will be useful not only during the fight against the epidemic of coronavirus infection, but also in other emerging diseases.

To describe the information technology of intelligent information system of epidemiological diagnostics the UML 2.5 was chosen.

It has a perceivable and expressive visual modeling language, specially designed for the development and documentation of models of complex systems for various

---

[1]Continued from No. 3, 2022.

purposes, which allows, in the context of the system being developed, to simultaneously achieve not only the universality of the representation of models for a variety of applications, but also the possibility of describing rather subtle details of the implementation of these models in relation to specific systems.

The original concepts of the UML language have the ability to expand and specialize for a more accurate representation of systems models in a specific subject area. The description of the UML itself contains a mechanism for extending basic concepts, which is an independent element of the language and has its own description in the form of extension rules. At the same time, overriding the basic concepts of the language should be avoided for whatever reason. This can lead to ambiguous interpretation of their semantics and possible confusion.

UML description supports a model specification that is independent of specific programming languages and software systems design tools — none of the UML constructs should depend on the specifics of its implementation in known programming languages. One of the components of the system being developed is implemented in the Python language for modeling the predicted values of the incidence in the form of a dedicated server, therefore, a flexible description of the interaction of the system components is required.

The aim of this article is to discuss the developed intelligent information system for epidemiological diagnostics with an emphasis on the modern information technologies used.

The structure of the paper is constructed as follows. Section 1, namely current research analysis, provides briefly the overview of current state of epidemic process models and intelligent information systems development approaches. Section 2 describes the user interaction with an intelligent information system of epidemiological diagnostics. Section 3 discusses the aspects of the implementation of the component architecture of the application. Conclusions describe outcomes of proposed information technology.

1. THE CURRENT RESEARCHES ANALYSIS

Due to the current pandemic of COVID-19 number of researchers investigating the epidemic processes has increased. But the transfer of methods and models from other domains has led to shortcomings of methods and models of epidemic processes. Many scientists do not take into account the specifics of the spread of epidemic morbidity, which includes multiple transmission routes, the heterogeneity of the population and the environment, the changing virulence of infection, etc.

The paper [3] presents the results of the EpiGraph model, an epidemic simulator extended to simulate the spread of COVID-19. The model is applied to the metropolitan area of Madrid. The disadvantage of the model is that the authors used a contagiousness index calculated for another territory, and does not take into account the peculiarities of human interaction and the spread of the virus in Spain. The authors of the article [4] conducted a systematic computational fluid dynamics-based investigation of indoor airflow and the associated aerosol transport in a restaurant setting, where likely cases of airflow-induced infection of COVID-19 caused by asymptomatic individuals were widely reported by the media. Despite the interesting approach authors have not taken into account the countermeasures such as masks, etc., that decrease the speed of infection propagation. A. Msalmi et al. performed long-term forecasting of COVID-19 infection in real-time using SEIR and logistic growth models [5]. The drawback of that research is that we do not need a long-term forecast of the epidemic process, because according to the short-term forecasts decision-makers

should apply immediate counter measures which will influence further dynamics of the morbidity propagation. The Poisson autoregression model of the daily new observed cases is presented in [6]. The model can reveal whether contagion has a trend, and where is each country on that trend. The disadvantage of the proposed approach is the impossibility of identification of factors influencing the dynamics of infection. A.A. Vagis, A.M. Gupal and I.V. Sergienko have developed procedures for determining mutations and their location in gene sequences which allow solving the following important problems: to conduct a detailed statistical analysis (including for age groups of patients) in relation to the number of mutations in encoding gene regions (exons) and in introns, as well as to confirm ahypothesis about protecting mechanisms in introns [7]. P.S. Knopov and A.S. Korkhin have proposed the stepwise solution of the problem of the dynamics of coronavirus cases in the form of a switching regression whose switching points are unknown [8]. The line of regression was obtained in the form of a piecewise linear function of time since the ends of these intervals coincide with the last switching points of the previous interval plus one or with the ends of the observation interval

E. Malkov uses a Susceptible-Exposed-Infectious-Resistant-Susceptible compartment model with the time-varying transmission rate to investigate the scenarios with possibility of reinfection of COVID-19 [9]. In [10] a sophisticated extension of a classical SEIR model, the simulation tool CovidSIM Version 1.0 has been presented. N. Ghaffarzadegan discusses what-if analysis for controlling the spread of COVID-19 in universities in U.S. using SEIR compartment model [11]. The paper [12] describes SQUIDER model incorporates additional processes into the classic SIR model by several new compartments, which we will denote as U (Undetected infected), E (undetected recovered) and Q (pseudo-Quarantine, a bin to hold a segment of the susceptible and undetected infected populations allowing us to model reduced human interactions due to social distancing). [13] proposes the compartment model with six subpopulations, which are: susceptible ($S(t)$), exposed ($E(t)$), infected ($I(t)$), asymptomatic ($A(t)$), recovered ($R(t)$) and dead ($D(t)$) for simulation the COVID-19 dynamics in Mexico.

The main disadvantage of studies [9–13] is that application of classic compartment models based on differential equations does not take into account the intelligent behavior of people as sources of infection. Another disadvantage is that systems of differential equations are complicated and it is hard to implement changes to calibrated models caused by changing infection virulence changing. That disadvantages can be eliminated using proposed compartments with agent-based approach to epidemic process simulation such as shown in [14–18].

Despite big number of researches in simulation of epidemic processes there are lack of developed information systems for decision-making in epidemiological diagnostics. The research [19] describes that using information systems is very important for developing countries for access to health care is a significant factor that contributes to a healthy population. Paper describes the methodological approach for the development of a real time electronic health record, based on the statistical and geographic information for the identification of various diseases and accidents that can happen in a specific place. But the proposed system does not have any tools for morbidity data analysis. The paper [20] describes geo-information system for public health, but proposed information system does not allow simulation and investigation of morbidity data. A. Amadoz and González-Candelas F. offer the epiPATH information system for the storage and management of molecular epidemiology data [21]. The proposed database covers many aspects of sample sources, samples,

laboratory processes, molecular sequences, phylogenetics results, clinical tests and results, clinical information, treatments, pathogens, transmissions, outbreaks and bibliographic information. Communication between end-users and the selected Relational Database Management System is carried out by default through a command-line window or through a user-friendly, web-based interface which provides access and management tools for the data. Still there are no possibilities to investigate the morbidity within the proposed system.

However, by today many approaches to development the intelligent information systems are proposed by different researches all over the world. The best practices can be scaled for epidemiology and Public Health domain. V.Y. Meytus has discussed the general principles and methods underlying the creation of intelligent information systems, namely: intelligent modeling, which is part of the definition of intelligence and shows how intelligent systems model the domain [22]. The areas of application of existing solutions, the way of their implementation, and unique features that can be useful in solving the problem of placement of communications have been considered in [23]. In the paper [24] information and communication technologies and IT-based instrumentation and control systems, hardware and software components are analyzed in context of "green" paradigm, what is important in current paradigm of sustainable development. Iu.V. Krak, G.I. Kudin and A.I. Kulyas have proposed the method for multidimensional information scaling based on the results of the theory of perturbation of pseudoinverse and projection matrices and solutions of systems of linear algebraic equations [25]. Proposed approach allows to perform a preliminary analysis of information with a considerable number of hardly separable classes. Paper [26] provides a general description of the typical methods that are used in multicriteria decision-making to determine the values of the coefficients of importance of indicators that characterize a composite system: the analytic hierarchy process, the methods of critical distance, of pairwise comparison, and of rank, the Fishburn, the CRITIS, and the entropy methods. V.I. Gritsenko and A.A. Ursatiev have discussed methods and approaches of using cloud technologies in medical data storage, processing and transferring [27].

Taking into account all the drawbacks and positive experience of developed epidemic processes models and intelligent information systems based on architecture design described in [1] the information technology of epidemiological diagnostics is proposed.

2. USER INTERACTION WITH AN INTELLIGENT INFORMATION SYSTEM

Developed intelligent information system of epidemiological diagnostics consists of four subsystems, each of which solves different tasks and contain different methods and models:

1. Forecasting subsystem is building the forecasted dynamics of epidemic process using different methods and models:
— making the forecast using the neural network models;
— making the forecast using the random forest method;
— making the forecast using the nearest neighbor method;
— making the forecast using lasso regression;
— making the forecast using linear regression;
— making the forecast using the gradient boosting method;
— making the forecast using ridge regression.

2. Population model subsystem is revealing the rules of the epidemic process dynamics using different methods and models and solves the following tasks:

— identification of factors influencing the epidemic process;

— identification of factors that do not affect the epidemic process;

— identification of risk groups;

— identification of the necessary immune layer;

— identification of the sources of infection;

— identification of the effective control measures;

— providing experimental study with epidemic process dynamics.

3. Data visualization and generating reports subsystem solves the following tasks:

— forecast visualization;

— creating reports in subsystem;

— visualization of population dynamics models.

4. Recommendation system subsystem is playing the following roles in intelligent information system:

— introducing of expert solutions into the subsystem;

— viewing recommendations in the system.

The use cases diagram is presented in Fig. 1.

Use case diagram is used to describe a set of actions (use cases) that a system or subsystems, as subjects, should or can perform in collaboration with one or more external users of the system. Each use case should provide some observable and valuable result for the participants or other interested parties in the system.
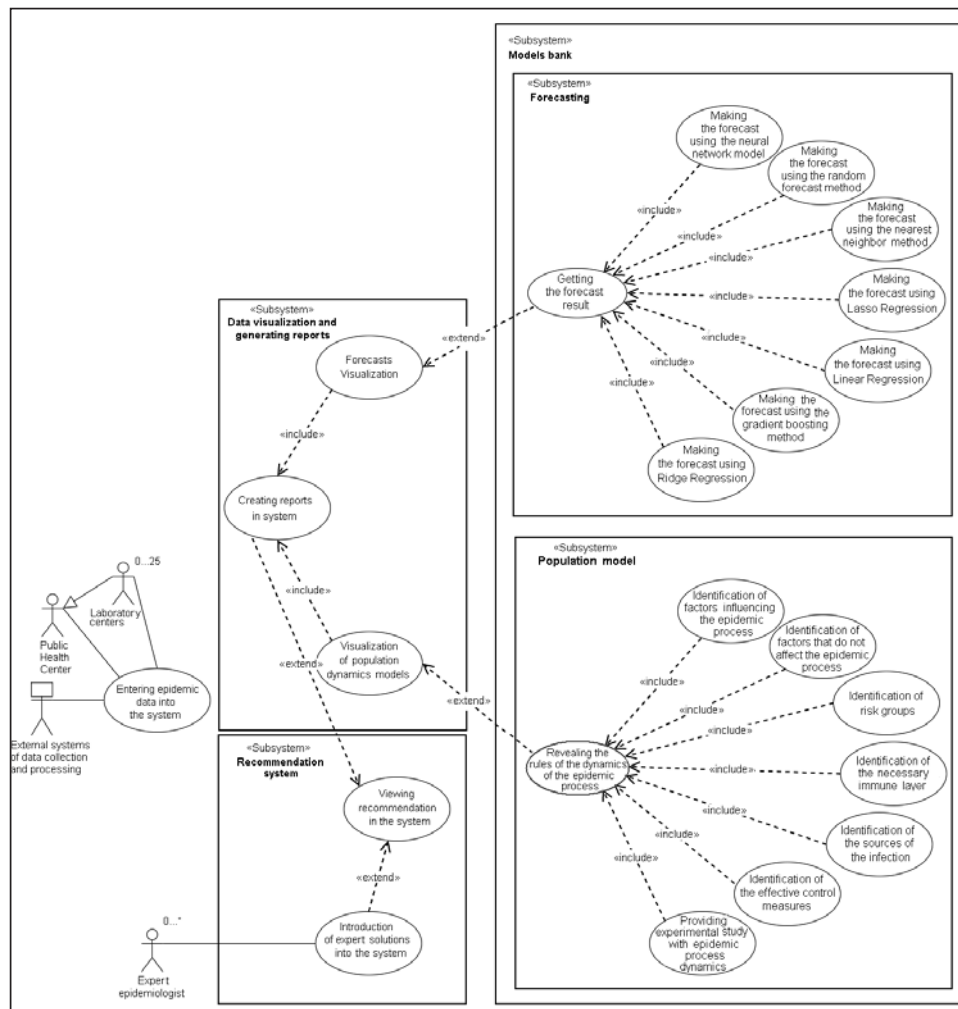

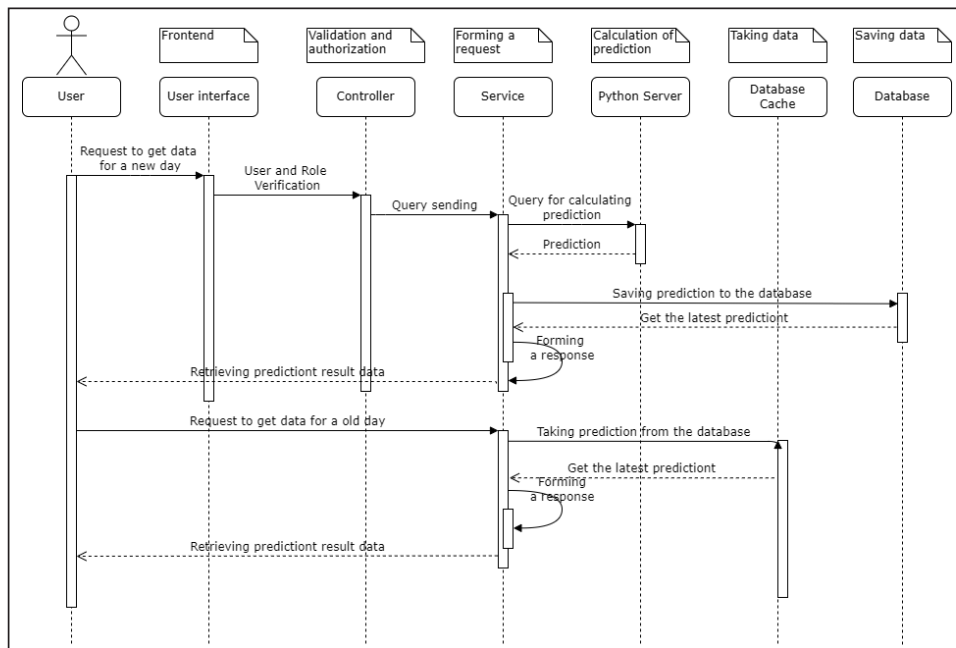*Fig. 1.* Use cases diagram of intelligent information system

Fig. 2. Sequence diagram of intelligent information system

Only binary associations are allowed between actors and use cases, using plurality at one or both ends of the link for descriptive detail. The diagram uses Extend relationships — a directed relationship that indicates how and when a behavior defined in a normally optional extension use case can be inserted into a behavior defined in an extended use case. It also uses an Include relationship, a directed relationship between two use cases that is used to demonstrate that the behavior of an included use case (append) is inserted into the behavior of an inclusive (base) use case.

For all options for using the system, sequence diagrams have been developed (one of which is shown in Fig. 2) — a type of interaction diagram that focuses on the exchange of messages between several life lines of objects of the described system. A sequence diagram describes the interaction by focusing on the sequence of messages being exchanged, together with the corresponding specifications of the appearance on the lifelines.

Figure 2 describes sequence diagram of the intelligent information system. Sequence diagram is the most common kind of interaction diagram, which focuses on the message interchange between a number of lifelines. Sequence diagram describes an interaction by focusing on the sequence of messages that are exchanged, along with their corresponding occurrence specifications on the lifelines.

To generate a user request to receive a forecast on a new day, the user must log in and authenticate in the system to determine the authority, then a request is sent to the service. The service sends a request to an external server with a prediction algorithm, then we get the prediction result. We store the forecasting result in the database by today's date. We form a response to the user on the service, and send the response to the user interface.

Forming a user's request for a forecast for the past days is no different from a request for a new day, except that we do not form a request to an external server with a forecasting algorithm, but immediately refer to the database to obtain the forecast result for the desired date.

Sequence diagram consists of the following elements:

— *User* is a user of a software product (in this case, a website);

— *User Interface* is user interface (in this case, a window in the browser); needed for user interaction with the software product;

— *Controller* is a place (class by methods) to receive requests from users of a software product (website);

— *Service* is the place where the requests of users of the software product (website) are processed, it also generates a response for the user after contacting the Database;

— *Python Server* is a place to receive and send disease forecasts;

— *Database Cache* is a place to store frequently requested data;

— *Database* is a place to store user data, statistics and forecasts.

3. ASPECTS OF THE IMPLEMENTATION OF THE COMPONENT ARCHITECTURE OF THE APPLICATION

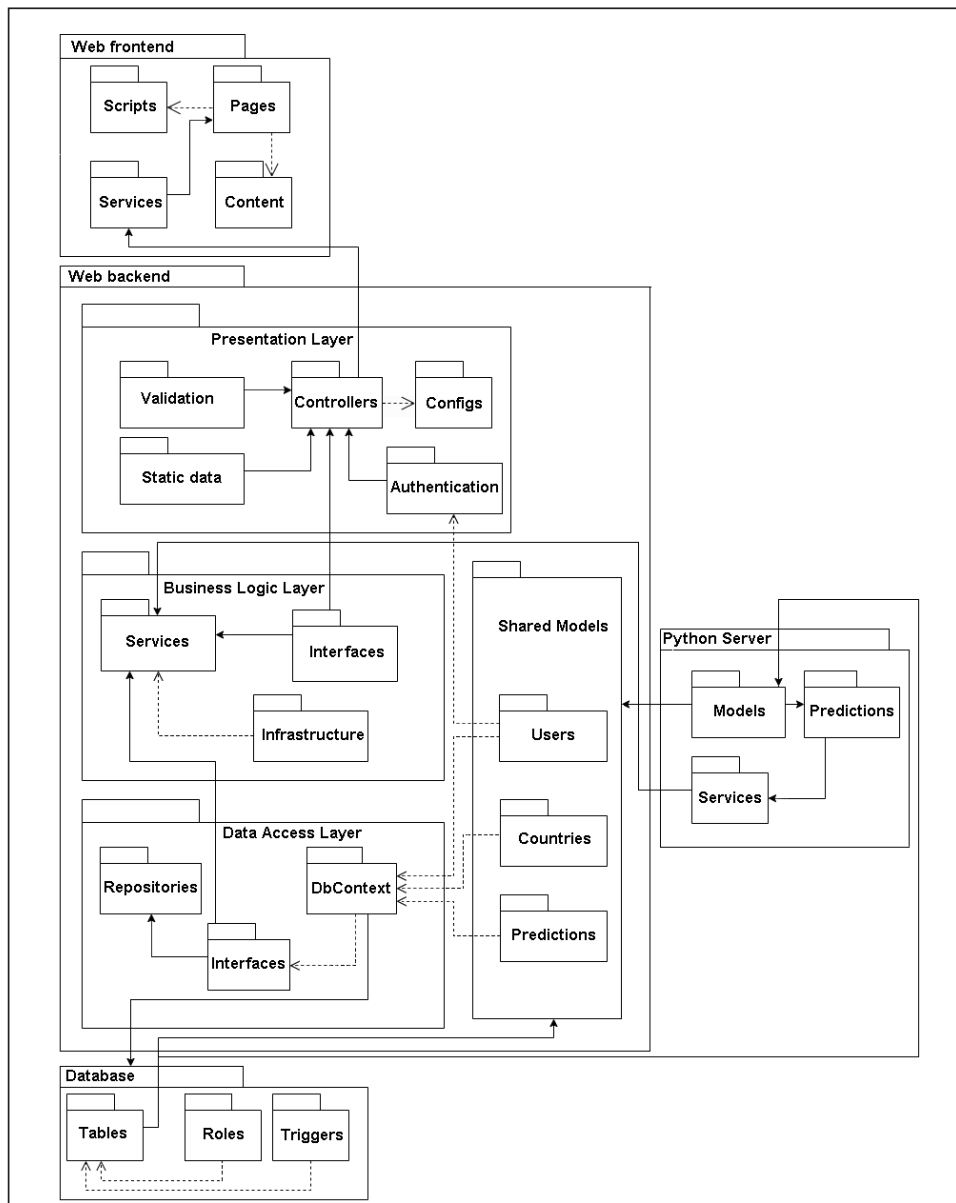Figure 3 describes the component diagram of the information system.

Fig. 3. Diagram of components of information system

Component diagram shows components, provided and required interfaces, ports, and relationships between them. This type of diagrams is used in Component-Based Development to describe systems with Service-Oriented Architecture. Component-based development is based on assumptions that previously constructed components could be reused and that components could be replaced by some other "equivalent" or "conformant" components, if needed. The artifacts that implement component are intended to be capable of being deployed and re-deployed independently, for instance to update an existing system. Components in UML can represent both logical components and physical components, along with the artifacts that implement them and the nodes on which they are deployed and run.

ASP.NET MVC has an Application Model that represents the components of an MVC program. Reading and processing this model allows you to change the behavior of the MVC elements. By default, MVC obeys certain rules to determine which classes are considered controllers, which methods of those classes are actions, and how parameters and routing work. This behavior was customized according to the program's needs by creating our own conventions and applying them globally or as attributes.

The following elements of the diagram presented in Fig. 1 are included:

— *Shared Models* is a part of the project structure that describes the models used in other project structures, *Users* participate in the authorization process, *Countries and Predictions* models participate in the process of saving to the database and executing the forecasting algorithm. Throughout the house of models created at different levels of the project architecture, you can quickly and structurally transfer data between them, following the SOLID rules and without breaking the dependencies between the project components.

— *Data Access Layer* is a part of the project structure containing the connection to the database, *Interfaces* and their implementation in *Repositories* are used to manipulate the database. In order to connect to the database, the repository uses a special object — the database context, which directly connects to the database and allows us to get the information we need.

— *Business Logic Layer* is a part of the project structure that describes the behavior of the user of requests for obtaining forecast data, if necessary, it sends a request to an external server to calculate the forecast, then saves it to the database context. This layer is the center of concentration of almost all the logic of the project, and here the data transformation from the types of models of one architectural layer to another takes place. Thanks to this, we can easily manipulate the data and transform it into the format we need.

— *Presentation Layer* is a part of the project structure, which includes the process of validation and user authentication, as well as in this part of the project there are static data and configs for setting up the project. This layer is also responsible for reproducing the received information from the previous two layers in a directly readable format (for example, on a browser page).

An association (solid arrow) indicates that objects of one entity (class) are connected to objects of another entity in such a way that you can move from one class to another. This is a common case of composition and aggregation.

Dependency (dashed arrow) means there is such a relationship between classes that changing the specification of the provider class can make the dependent class work, but not vice versa.

Figure 4 shows deployment diagram, which is specialized realization dependency used to (optionally) define classifiers that realize the contract offered by a component in terms of its provided interfaces and required interfaces.
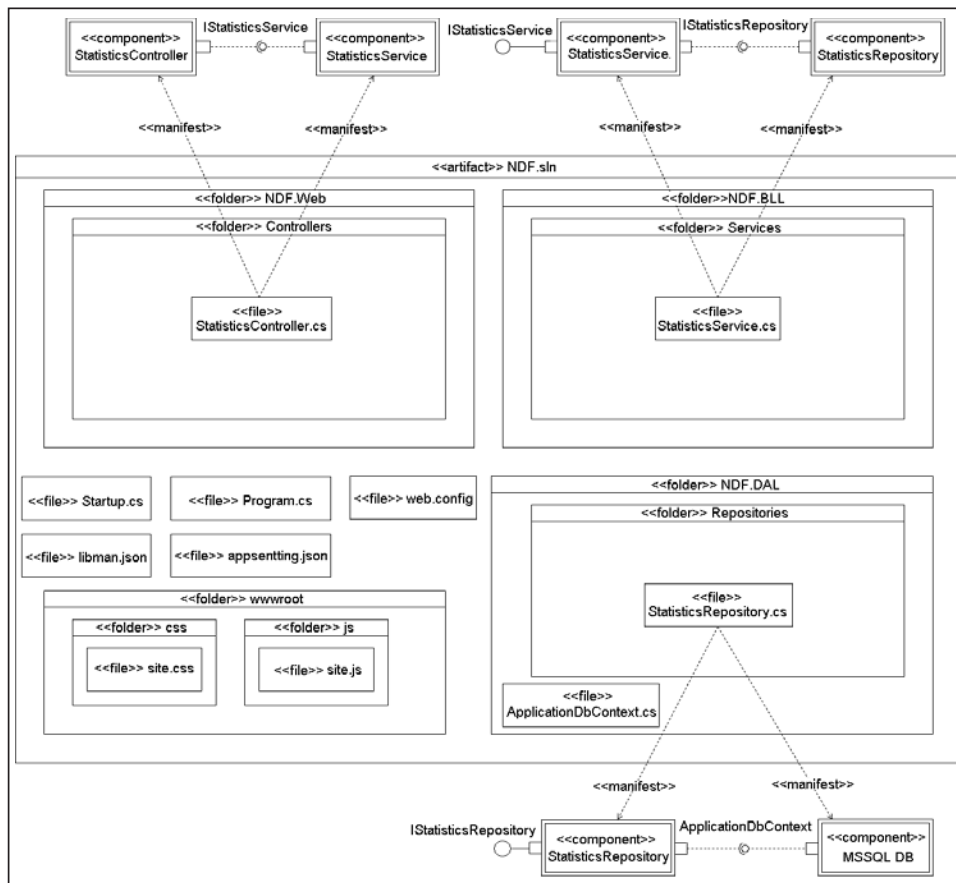
Fig. 4. Deployment diagram

A deployment diagram is a structural diagram that shows the architecture of a system as the deployment (distribution) of software artifacts across deployment targets. Artifacts are specific elements of the physical architecture that are the result of the development process. Examples of artifacts are executables, libraries, databases, configuration files, and so on. A deployment target is usually represented by a node, which is either a hardware device or some kind of software runtime. Thenodes are connected by means of communication channels to create network systems of arbitrary complexity.

The components have been deployed directly to the nodes in the deployment diagrams. Artifacts are deployed to nodes and artifacts can manifest (implement) components. Components are deployed to nodes indirectly through artifacts. Deployment diagrams describe the architecture at the specification level. The specification shows some overview of deployment from artifacts to deployment targets, without reference to specific examples of artifacts or nodes.

The following elements were included:

— <<artifact>> NDF.sln is the main project file;

— <<folder>> NDF.Web is the folder with the first architectural layer (client side);

— <<folder>> NDF.BLL is the folder with the second architectural layer (business logic of the project);

— <<folder>> NDF.DAL is the folder with the third architectural layer (level of access to data from the database);

— <<folder>> Controllers is the folder with controllers (router by links);

— <<folder>> Services is the folder with services (contains the logic of project methods);

— <<folder>> Repositories is the folder with repositories (contains logic for communicating with the database context);

— <<file>> StatisticsController.cs is the controller responsible for returning forecast and statistics;

— <<file>> StatisticsService.cs is the service responsible for processing the received forecast and statistics;

— <<file>> StatisticsRepository.cs is the service responsible for getting forecast and statistics from the database context;

— <<file>> ApplicationDbContext.cs is the database context (through it, direct connection with the database is carried out);

— <<file>> Startup.cs is the file responsible for project settings (setting dependencies between classes, setting up authorization / authentication);

— <<file>> Program.cs is entry point into the application;

— <<file>> web.config is the file responsible for setting the project configuration;

— <<file>> libman.json is file responsible for configuring third-party libraries;

— <<file>> appsettings.json is file responsible for setting application variables;

— <<folder>> wwwroot is the folder where the project's static data is stored;

— <<folder>> css is the the folder where the styles of the website are stored;

— <<folder>> js is folder where website scripts are stored;

— <<file>> site.css is the file stores the main styling of the website;

— <<file>> site.js is the file contains the main scripts of the website.

CONCLUSIONS

Based on the architecture proposed in [1], the information system of epidemiological diagnostics has been designed. The developed information system is open and can be supplemented with models and methods of epidemiological diagnostics. As part of the study, a comprehensive analysis of the current state of information systems design and model and methods of epidemiological diagnostics has been carried out. The drawbacks of the existing information systems of epidemiological diagnostics are revealed. The positive experience in the development of intelligent information systems has been extended to the subject area of epidemiology and public health. The user interaction with an intelligent information system has been discussed. Sequence diagram of the intelligent information system, which focuses on the message interchange between a number of lifelines and describes an interaction by focusing on the sequence of messages that are exchanged, along with their corresponding occurrence specifications on the lifelines have been developed. Use case diagram has been used to describe a set of actions (use cases) that a system or subsystems, as subjects, should or can perform in collaboration with one or more external users of the system. The aspects of the implementation of the component architecture of the application have been shown. The component diagram shows components, provided and required interfaces, ports, and relationships between them. The components have been implemented directly to the nodes in the deployment diagram, that shows the architecture of an information technology as the distribution of software artifacts across deployment targets. The architecture and intelligent information technology of epidemiological diagnostics proposed in the framework of the study allows collecting, processing, analyzing and researching information on epidemic processes and infectious diseases. Information technology contains 4 subsystems with banks of models and methods of epidemiological diagnostics. The open architecture makes it possible to supplement the system with up-to-date models and methods of researching epidemic processes that are relevant at a certain moment in a certain territory.

Future research prospects are the improvement of the integration of external models, methods and information technologies into the proposed one. It is planned to introduce an information system in medical and preventive institutions of Ukraine for testing in real time based on existing data on epidemic morbidity.

REFERENCES

1. Bazilevych K.O., Chumachenko D.I., Hulianitskiy L.F., Meniailov I.S, Yakovlev S.V. Intelligent decision support system for epidemiological diagnostics. I. A concept of architecture design. *Кібернетика та системний аналіз.* 2022. Т. 58, № 3. С. 30–41.

2. Yakovlev S., Bazilevych K., Chumachenko D., Chumachenko T., Hulianytskyi L., Meniailov I., Tkachenko A. The concept of developing a decision support system for the epidemic morbidity control. *Proc. 3rd International Conference on Informatics & Data-Driven Medicine* (19–21 November, 2020, Växjö, Sweden). Växjö, 2020. Vol. 2753. P. 265–274.

3. Singh D.E., Marinescu M.C., Guzmán-Merino M., Durán C., Delgado-Sanz C., Gomez-Barroso D., Carretero J. Simulation of COVID-19 propagation scenarios in the Madrid metropolitan area. *Frontiers of Public Health.* 2021. Vol. 9, 636023. https://doi.org/10.3389/ fpubh.2021.636023.

4. Liu H., He S., Shen L., Hong J. Simulation-based study of COVID-19 outbreak associated with air-conditioning in a restaurant. *Physics of Fluids.* 2021. Vol. 33 (2), 023301. https://doi.org/10.1063/5.0040188.

5. Msmali A., Zico M., Mechai I., Ahmadini A. Modeling and simulation: a study on predicting the outbreak of COVID-19 in Saudi Arabia. *Discrete Dynamics in Nature and Society*. 2021. Vol. 2021, 5522928. https://doi.org/10.1155/2021/5522928.

6. Agosto A., Giudici P. A poisson autoregressive model to understand COVID-19 contagion dynamics. *Risks*. 2020. Vol. 8 (3), 77. https://doi.org/10.3390/risks8030077.

7. Vagis A.A., Gupal A.M., Sergienko I.V. Determination of risk groups for the Covid-19 underlying deseases. *Cybernetics and Systems Analysis*. 2021. Vol. 57, N 2. P. 223–227. https://doi.org/10.1007/s10559-021-00347-9.

8. Knopov P.S., Korkhin A.S. Statistical analysis of the dynamics of coronavirus cases using stepwise switching regression. *Cybernetics and Systems Analysis*. 2020. Vol. 56, N 6. P. 943–952. https://doi.org/10.1007/s10559-020-00314-w.

9. Malkov E. Simulation of coronavirus disease 2019 (COVID-19) scenarios with possibility of reinfection. *Chaos, Solitons and Fractals*. 2020. Vol. 139, 110296. https://doi.org/10.1016/ j.chaos.2020.110296.

10. Schneider K.A., Ngwa G.A., Schwehm, M., Eichner L., Eichner M. The COVID-19 pandemic preparedness simulation tool: CovidSIM. *BMC Infectious Diseases.* 2020. Vol. 20, 859. https://doi.org/10.1186/s12879-020-05566-7.

11. Ghaffarzadegan N. Simulation-based what-if analysis for controlling the spread of Covid-19 in universities. *PLoS ONE.* 2021. Vol. 16 (2), e0246323. https://doi.org/10.1371/journal.pone.0246323.

12. Khan Z., Van Bussel F., Hussain F. A predictive model for Covid-19 spread — with application to eight US states and how to end the pandemic. *Epidemiology and Infection*. 2020. Vol. 148, e249. P. 1–13. https://doi.org/10.1017/S0950268820002423.

13. Avila-Ponce de Leon U., Perez A.G.C., Avila-Vales E. An SEIARD epidemic model for COVID-19 in Mexico: Mathematical analysis and state-level forecast. C*haos, Solitons and Fractals.* 2020. Vol. 140, 110165. https://doi.org/10.1016/j.chaos.2020.110165.

14. Krak Iu., Barmak O., Radiuk P. Information technology for early diagnosis of pneumonia on individual radiographs. *Proc. IDDM'2020: 3rd International Conference on Informatics & Data-Driven Medicine* (19–21 November, 2020, Växjö, Sweden). Växjö, 2020. Vol. 2753. P. 11–21.

15. Gharakhanlou N.M., Hooshangi N. Spatio-temporal simulation of the novel coronavirus (COVID-19) outbreak using the agent-based modeling approach (case study: Urmia, Iran). *Informatics in Medicine Unlocked.* 2020. Vol. 20, 100403. https://doi.org/10.1016/j.imu.2020.100403.

16. Kerr C.C., Stuart R.M., Mistry D., Abeysuriya R.G., Rosenfeld K., Hart G.R., et al. Covasim: An agent-based model of COVID-19 dynamics and interventions. *PLoS Computational Biology.* 2021. Vol. 17 (7), e1009149. https://doi.org/10.1371/journal.pcbi.1009149.

17. Cuevas E. An agent-based model to evaluate the COVID-19 transmission risks in facilities. *Computers in Biology and Medicine*. 2020. Vol. 121, 103827. https://doi.org/10.1016/j.compbiomed.2020.103827.

18. Truszkowska A., Behring B., Hasanyan J., Zino L., Butail S., Caroppo E., Jiang Z.-P., Rizzo A., Porfiri M. High-resolution agent-based modeling of COVID-19 spreading in a small town. *Advanced Theory and Simulations*. 2021 Vol. 4, Iss. 3. 2000277. https://doi.org/10.1002/adts.202000277.

19. Diomidous M., Pistolis J., Mechili A., Kolokathi A., Zimeras S. Healthcare information Systems for the epidemiologic surveillance within the community. *Studies in Health Technology and Informatics.* 2013. Vol. 190. P. 252–254.

20. Gómez-Rubio V., Ferrándiz-Ferragud J., López-Quílez A. Epidemiological Information Systems. In: Interfacing Geostatistics and GIS. Pilz J. (Ed.). Berlin; Heidelberg: Springer-Verlag, 2009. P. 235–248. https://doi.org/10.1007/978-3-540-33236-7_18.

21. Amadoz A., González-Candelas F. epiPATH: an information system for the storage and management of molecular epidemiology data from infectious pathogens. *BMC Infectious Diseases.* 2007. Vol. 7, 32. https://doi.org/10.1186/1471-2334-7-32.

22. Meytus V.Yu. Problems of constructing intelligent systems. Intelligent modeling. *Cybernetics and Systems Analysis.* 2021. Vol. 57, N 4. P. 509–520. https://doi.org/10.1007/s10559-021-00376-4.

23. Vdovichenko O., Perepelitsyn A. Technologies for building systems of remote lining of communication lines: a practical example of implementation. *Radioelectronic and Computer Systems*. 2021. Vol. 98 (2). https://doi.org/10.32620/reks.2021.2.03.

24. Kharchenko V., Gorbenko A., Sklyar A., Phillips C. Green computing and communications in critical application domains: Challenges and solutions. *Proc. International Conference on Digital Technologies 2013* (29–31 May 2013, Zilina, Slovakia). Zilina, 2013. P. 191–197. https://doi.org/10.1109/DT.2013.6566310.

25. Krak Iu.V., Kudin G.I., Kulyas A.I. Multidimensional scaling by means of pseudoinverse operations. *Cybernetics and Systems Analysis*. 2019. Vol. 55, N 1. P. 22–29. https://doi.org/10.1007/s10559-019-00108-9.

26. Potomkin M.M., Sedlyar A.A., Deineha O.V., Kravets O.P. Comparison of the methods used in multicriteria decision-making to determine the values of the coefficients of importance of indicators that characterize a complex system. *Cybernetics and Systems Analysis*. 2020. Vol. 56, N 6. P. 990–999. https://doi.org/10.1007/s10559-020-00319-5.

27. Gritsenko V.I., Ursatiev A.A. Modern information technologies of data storage and computing. Kyiv: Nauk. Dumka, 2017. 213 p. (in Russian).

К.О. Базілевич, Д.І. Чумаченко, Л.Ф. Гуляницький,
Є.С. Меняйлов, С.В. Яковлев
ІНТЕЛЕКТУАЛЬНА СИСТЕМА ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ ДЛЯ ЕПІДЕМІОЛОГІЧНОЇ ДІАГНОСТИКИ. ІІ. ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Анотація. У статті запропоновано компоненти інтелектуальної системи підтримки прийняття рішень для епідеміологічної діагностики та досліджено їхню взаємодію з користувачем. До складових системи включено банк моделей та методів машинного навчання, банк моделей популяційної динаміки, засоби візуалізації і формування звітів, блок формування управлінських рішень. Надано загальну концепцію інформаційної технології для гарантування біобезпеки населення. Розроблено модель варіантів використання зазначеної інформаційної технології користувачем та побудовано діаграму послідовностей. Запропоновано модель компонентів інформаційної технології та шляхи їх розгортання на сервері.

Ключові слова: система підтримки прийняття рішень, інформаційна технологія, епідеміологічна діагностика, машинне навчання, популяційна динаміка.