



WROCŁAWSKA WYŻSZA SZKOŁA
INFORMATYKI STOSOWANEJ

Biuletyn Naukowy
Wrocławskiej Wyższej Szkoły
Informatyki Stosowanej

INFORMATYKA

TOM 9. NUMER 1
WROCŁAW 2022

KOLEGIUM REDAKCYJNE (EDITORIAL BOARD)

Redaktor Naczelny (Editor-in-Chief)

Piotr Kardasz

Z-ca Redaktora Naczelnego (Deputy Editor-in-Chief)

Mirosław Dela

Redaktor Merytoryczny (Content Editor)

Łukasz Jeleń

Redaktor Statystyczny (Statistical Editor)

Ryszarda Getko

Redaktor Językowy (Language Editor)

Jakub Zbądzki

Sekretarz Wydawniczy (Assistant Editor)

Marcin Radzewicz

RADA NAUKOWA (EDITORIAL ADVISORY BOARD)

Radostin Dimitrov – Technical University of Varna

Veselin Mihaylov – Technical University of Varna

Olena Mitryasova – Petro Mohyla Black Sea National University

Volodymyr Pohrebennyk – Lviv Polytechnic National University

Tibor Vince – Technical University of Kosice

Jurij Warecki – Lviv Polytechnic National University

Adres Redakcji (Editorial Office)

Wrocławska Wyższa Szkoła Informatyki Stosowanej, Wydział Informatyki,

ul. ks. M. Lutra 4, 54-239 Wrocław, Poland

tel. (48) (71) 788-94-25

fax. (48) (71) 799-19-37

e-mail: wydawnictwo@wwsis.pl

<https://wydawnictwo.horyzont.eu/>

© Copyright by Wrocławska Wyższa Szkoła Informatyki Stosowanej, Wrocław 2022

ISSN 2082-9892

Skład i łamanie (Typesetting): Marcin Radzewicz

Printed by Wrocławska Wyższa Szkoła Informatyki Stosowanej, Wrocław, Poland

Redakcja *Biuletynu Naukowego Wrocławskiej Wyższej Szkoły Informatyki Stosowanej. Informatyka* informuje, że wersją pierwotną czasopisma jest wydanie papierowe.

Spis treści • Contents

RECENZOWANE ARTYKUŁY NAUKOWE PEER-REVIEWED SCIENTIFIC ARTICLES

Piotr Kardasz, Marcin Skocz, Ewa Kardasz, Piotr Jednaszewski

Logistyka informacji – najcenniejsza w XXI wieku4

Information logistics – the most valuable commodity of the 21st century

Mirosław Dela, Piotr Kardasz

Treść roszczenia o usunięcie skutków naruszenia dóbr osobistych w internecie9

The content of the claim for the elimination of the consequences of infringement of personal rights on the Internet

Beata Laszkiewicz, Tomasz Sobczak

Analiza i implementacja algorytmów odnajdywania ścieżki do zastosowania w grach przeglądarkowych.....6

Analysis and implementation of pathfinding algorithms for use in browser games

Jakub Zbądzki, Anna Rogowska-Wandowicz

Testowanie użyteczności oprogramowania do nauki języków klasycznych. Case study aplikacji mobilnej *Speak Latin: Learn Latin Language Offline*.....25

Usability testing of software for learning classical languages. Case study of the mobile application Speak Latin: Learn Latin Language Offline

RECENZOWANE ARTYKUŁY POPULARNONAUKOWE PEER-REVIEWED POPULAR SCIENCE ARTICLES

Konrad Szynalski, Dawid Różański

Wprowadzenie do modelowania w języku UML.....31

Introduction to UML modeling

Drodzy Czytelnicy!

Wraz z nowym numerem *Biuletynu Naukowego Wrocławskiej Wyższej Szkoły Informatyki Stosowanej. Informatyka* inicjujemy wiele zmian, które wpłyną pozytywnie na dostęp do artykułów i umiędzynarodowienie czasopisma. Od teraz poszczególne teksty będą opatrzone numerami ORCID, a także publikowanie indywidualnie na stronie Wydawnictwa WWSIS wraz z odpowiednimi metadanymi, co pozwoli na ich efektywne indeksowanie, a autorzy publikacji będą zawierać umowy na zapewniającej maksymalną otwartość licencji CC BY 4.0.

W gospodarce 4.0 systematycznie wzrasta rola logistyki informacji, już dziś uznawanej za jeden z elementów szczególnie mocno wpływających na innowacyjność i konkurencyjność przedsiębiorstw. Artykuł *Information Logistics – the Most Valuable Commodity of the 21st Century* prezentuje ją jako nową dziedzinę nauki, niezwykle korzystną dla biznesu, ale wymagającą też szczególnej uwagi ze względu na powszechnie występujące w internecie formy dezinformacji, w których wykorzystuje się m.in. algorytmy i boty.

Sieć jest także miejscem, gdzie łatwo o nadużywanie anonimowości. Tekst *The content of the claim for the elimination of the consequences of infringement of personal rights on the Internet* prezentuje ten problem w praktycznym ujęciu, wskazując, jak pokrzywdzeni mogą złożyć odpowiednie oświadczenie w sądzie i na co powinni zwrócić uwagę, biorąc pod uwagę aktualne orzecznictwo i wiedzę z zakresu informatyki.

Artykuł *Analiza i implementacja algorytmów odnajdywania ścieżki do zastosowania w grach przeglądarkowych* prezentuje pogodniejszy fenomen związany z internetem, jakim są dostępne na wielu popularnych portalach i w serwisach społecznościowych gry. Autorzy wskazują optymalne algorytmy odnajdywania ścieżki, które można w nich zastosować, wraz z przykładem wykorzystania najbardziej efektywnego z nich w praktyce.

Coraz częściej edukacja przebiega w sposób niesformalizowany, indywidualny i cyfrowy. Nie wszystkie dostępne narzędzia okazują się jednak skuteczne. Artykuł *Testowanie użyteczności oprogramowania do nauki języków klasycznych. Case study aplikacji mobilnej Speak Latin: Learn Latin Language Offline* przedstawia tę kwestię z perspektywy informatycznej i językoznawczej, łącząc różne dyscypliny i perspektywy.

Na koniec zamieszczamy debiut absolwentów uczelni, artykuł *Wprowadzenie do modelowania w języku UML*, który omawia zagadnienie w klarowny i przystępny sposób. Diagramy UML ze względu na swoją czytelność i uniwersalność są powszechnie wykorzystywane w inżynierii oprogramowania, więc warto z nimi zapoznawać się jak najwcześniej w toku edukacji informatycznej.

Z życzeniami przyjemnej lektury

Redakcja *Biuletynu Naukowego Wrocławskiej Wyższej Szkoły Informatyki Stosowanej. Informatyka*

*Information logistics – the most valuable commodity of the 21st century***Logistyka informacji – najcenniejsza w XXI wieku****Piotr Kardasz¹, Marcin Skocz², Ewa Kardasz³, Piotr Jednaszewski⁴**

ABSTRACT: The first part of the article describes a new field of science, which is information logistics, and its expected role in the future. The overflow of information and the tools that create it are characterized: algorithms, bots, clickbaits, trolling and social media. Subsequently, the increasing importance of critical thinking in modern times was pointed out. The second part of the article describes the role of information logistics in an enterprise. The attention was paid to: the importance of systemic management of information flow in business, the conditions of an appropriate information flow in the enterprise, and the loss of information in the communication process. The third part of the article highlights the importance of information logistics in e-commerce companies. It also provides tips on how to take care of information logistics in e-commerce companies and characterizes the benefits of using information logistics in this type of enterprises.

STRESZCZENIE: W pierwszej części artykułu opisano nową dziedzinę nauki, jaką jest logistyka informacji, oraz jej przewidywaną rolę w przyszłości. Scharakteryzowano w niej nadmiar informacji i narzędzia, które go tworzą: algorytmy, boty, clickbait, trolling i social media. Następnie wskazano wzrastające współcześnie znaczenie krytycznego myślenia. W drugiej części artykułu opisano rolę logistyki informacyjnej w przedsiębiorstwie. Zwrócono uwagę na znaczenie systemowego zarządzania przepływem informacji w biznesie, warunki prawidłowego przepływu informacji w przedsiębiorstwie oraz utratę informacji w procesie komunikacji. W trzeciej części artykułu podkreślono znaczenie logistyki informacji w firmach e-commerce. Zawiera ona również wskazówki, jak zadbać o logistykę informacji w firmach e-commerce, oraz charakteryzuje korzyści płynące z zastosowania logistyki informacji w tego typu przedsiębiorstwach.

KEYWORDS: information, logistics, internet, e-commerce

SŁOWA KLUCZOWE : informacja, logistyka, internet, e-commerce

1. Information logistics in everyday life**1.1. Information overflow as one of the most popular phenomena on the Internet**

The 20th and 21st centuries is a time of dynamic development of civilization and technology. The automated world is built not only on many amenities, but also on threats. Such convenience is, for example, the Internet, which is a broadly understood source of knowledge available at any time and almost anywhere in the world. On the other hand, the threat is the surfeit of information (both valuable and completely worthless), with which – whether we like it or not – we deal with every day. Our decision-making often depends on who, when and with what content. The basic problem resulting from noticing information overload and its impact on us is finding answers to several questions⁵. The most important of them are as follows:

- How to distinguish between real information and fake news without time-consuming and laborious analysis?

- How to prevent social media from being used to influence public opinion, including ourselves?
- How to recognize the manipulation techniques constantly used in the network?
- How not to succumb to the above-mentioned manipulation techniques?
- How to reach only important and true information – both in life and in business?
- How to use this information in traditional business?
- How to use this information in the e-commerce business?

The answers to all of the above questions are provided below.

1.2. Information logistics – the answer to the problem of information overload

In the formulation of answers to the above questions, it

1. Wrocławska Wyższa Szkoła Informatyki Stosowanej we Wrocławiu, e-mail: pkardasz@horyzont.eu, ORCID 0000-0002-0933-3384.

2. Semper Avanti, e-mail: marcin.skocz@semperavanti.org, ORCID 0000-0002-9302-5762.

3. Fundacja Badań Rozwoju i Innowacji, e-mail: sekretariat@fbri.pl, ORCID 0000-0003-2038-8228.

4. St. Mary's Academy Ltd., e-mail: peterjednaszewski@gmail.com, ORCID 0000-0003-1061-907X.

5. Fake news, czyli jak manipulując informacją zdobyć nas w sieci, <http://fundacjawspomaganiawsi.pl/fake-news-czyli-jak-manipulujac-informacja-zdobyć-nas-w-sieci/> [access: 28.09.2021]

may be helpful to delve into a new, still unexplored field of knowledge, which is information logistics – the subject of this article. Information logistics is sometimes called the „most valuable commodity” for humanity in the 21st century. In order to properly select information, you need to be aware of the ways that web developers influence our decisions. These methods are mainly:

- algorithms,
- bots,
- clickbaits,
- trolling,
- social media (for example Facebook).

All of them are briefly described below.

1.3. Algorithms

The use of digital technologies is associated with leaving behind digital traces, and thus – being „tracked” by algorithms. What are algorithms? These are activities aimed at completing a task or solving a problem in the shortest possible time. Algorithms are used, among others, in the activities of the most popular Internet search engine, which is Google. For example, Google uses them to assess the usefulness of websites (the main factor is the quality of the content available on them) and try to understand the intentions of users who search for certain terms (these are keywords and key phrases). Algorithms, however, pose certain dangers. Firstly – owners of some websites attempt to manipulate ranking factors (for example, by saturating texts with frequently searched keywords and key phrases while „missing” the topic and the resulting disappointment of users with the content on the website). Secondly, they often collect and use information about us, not necessarily with our consent. How to avoid it? It is impossible to completely escape from algorithms.

1.4. Bots

Bots, also known as internet robots, are programs whose purpose is to perform specific actions on the web. These actions are typical of live users.

These can be activities such as:

- browsing websites,
- searching for information on websites,
- downloading files,
- issuing grades,
- writing comments,
- conducting conversations (to conduct conversations, which is a demanding task, special bots, i.e. chatbots and voicebots are used).

Bots now account for over 50 percent of all web traffic.

Therefore, it is worth bearing in mind that many ratings or opinions that we meet on the Internet are made not only by live users, but also bots.

1.5. Clickbaits

These are content whose primary purpose is to grab the audience’s attention. Typically, such content has a shocking, sensational title and / or thumbnail. Typical for click-

baits are:

- arousing curiosity, which is a natural human trait;
- awakening emotions and desires;
- spreading fake news (i.e. false or partially untrue information aimed at obtaining financial, political or prestigious benefits);
- disinformation;
- the intensity of graphics: colours, fonts and others;
- general „flashiness”.

Clickbaits often „offer” easy, fast and cheap (even free) solutions to problems, e.g.:

- „Check how to lose weight without exercising” (easy solution);
- „We know a way to earn a large amount of money in a few days” (quick solution);
- „Free English language course” (free solution).

They are usually in the form of exclamation points (purpose: shock, sensationalism) or interrogative sentences (purpose: arousing curiosity and emotions), e.g.:

- „It's unbelievable!”;
- „You just have to see it!”;
- „How did he do it?”;
- „How did it even happen?”

Why are clickbaits usually unsuccessful? There are at least several reasons. First – they often do not keep the promise contained in the title (obtaining the answer to the question). Second, they mislead the reader. Third, they contain manipulation, half-truths, or even lies. Fourth – being worthless information built on exaggeration and exaggeration, they simply waste our time and, in turn, cause irritation. Clickbaits are, of course, one of the most popular solutions that generate high traffic on popular websites that are full of them. However, anyone looking to source only high-quality content should avoid it⁶.

1.6. Trolling

Trolling is the behaviour of focusing the attention of network users on themselves, and then causing them irritation and anger. It is especially visible in chat rooms, internet forums and in those parts of the network where discussions are held (for example, YouTube). This behaviour is known as anti-social. People characterized by narcissism, Machiavellianism, psychopathy and sadism often admit that trolling is an extremely popular Internet activity.

1.7. Social media as a source of knowledge about us

It is also worth emphasizing that social media, such as Facebook, use knowledge about us for marketing purposes. How does it happen? An example is the initiative of scientists from the University of Cambridge, who several years ago sent a personality test to a huge number of Facebook users. Thus, Facebook obtained information about the preferences and tastes of many of its users. This information, in turn, allowed, among other things, to infer this:

6. *Clickbait – co to jest?*, <https://www.artefakt.pl/blog/epr/clickbait-co-to-jest> [access: 28.09.2021].

- what sexual orientation individual users have,
- what is their emotional profile,
- do they suffer from personality disorders, and if so – what (present),
- if they have anxiety, and if so, what are they caused by (past).

Many Facebook users did not complete the test, but they spread information about themselves daily by adding vacation photos, talking about changing jobs, liking fan pages about political and social topics or simply subscribing to certain groups and taking various activities on them. Of course, the more we know about us, the easier it is to reach us and manipulate us in turn. This is evidenced by the scandals regarding the legitimacy of Brexit and the 2017 US presidential election. Opponents of the UK's exit from the European Union and people who undermine the democratic nature of the US elections still talk about using information about Facebook users for propaganda and unethical purposes⁷.

1.8. How to deal with information overload, i.e. what is critical thinking

The human brain, when a person begins to think critically, i.e. not only absorb content, but also select and evaluate it, uses a lot of energy. Interestingly, the brain, which makes up just 2 percent of body weight, consumes about 25 percent of all our energy. This, in turn, causes fatigue and makes us want to „take shortcuts”, and thus – end the process of assimilating and half-memorizing, and give up selection. Such „shortcuts” only benefit the brain in a short time. In the long run, critical thinking:

- increases the number of neural connections, and in turn „develops” and „trains” our brain;
- arouses our curiosity, the satisfaction of which contributes to the feeling of happiness;
- slows down the ageing process.

1.9. What else prevents us from critical thinking, i.e. about emotions

The brain's tendency not to overexert is one of the two most important reasons that keep us from thinking critically. The second reason is the fact that very often emotions take over and make the reception of reality as it is difficult. This is because man by nature makes emotional, not rational, decisions. However, anyone who wants to be guided primarily by understand must constantly allow himself to think that his emotions are telling him not to make the best decisions.

1.10. „Think about how to think”, „think before you think”

These slogans are at the heart of critical thinking, which is and will certainly be one of the most desirable competences of the 21st century – both in everyday life and at work. How to think critically?

You need to:

- understand positions presented in texts;
- study and evaluate arguments, especially opposing ones;
- recognize persuasive and manipulative techniques;
- independently think about the problems described in the texts;
- create your positions supported by a thorough reading of the text and its structure;
- know and use the rules of logic.

2. Information logistics in the enterprise

2.1 Systemic management of information flow in business

The information we deal with daily includes manipulations, half-truths, and often lies. Appropriate selection of information is therefore a challenge for each of us. One more aspect related to information is worth emphasizing, namely – the management of the information system in the business. From the perspective of everyone – macro, small, but above all medium and large enterprises, system management of the information system is crucial for business development.

2.2. What is the information flow in the enterprise?

As Zbigniew Wąsik and Zbigniew Kotulski write in the article in the publication *Zarządzanie przedsiębiorstwem w warunkach konkurencji*, information flowing and collected in the enterprise becomes knowledge [...] necessary for making decisions, i.e. the so-called „know-how”, later it may constitute technical knowledge, which is a permanent asset of the enterprise, subject to protection as intellectual property⁸. However, in order for it to be as written by Wąsik and Kotulski, i.e. that the obtained data could constitute a solid basis for making the best possible decisions, the flow of information in the enterprise must meet certain, strictly defined conditions.

2.3. Conditions for the proper flow of information in the enterprise

Following Wąsik and Kotulski, there are seven basic conditions for the proper flow of information in an enterprise. They are described below.

- 1) The flow of information must be clear, transparent, understandable for each employee.
- 2) It should be very precise what each participant of the information flow is responsible for.
- 3) The company should introduce the standardization of the message, and thus – the flow of information – terms, signals, systems. This standardization will fully or partially exclude communication disruptions among employees.
- 4) It should be remembered that the transfer of information will take place in two ways – in a standardized manner

⁷ *Czy powinniśmy skasować Facebooka? Co tak naprawdę zrobiła Cambridge Analytica*, <https://technologia.dziennik.pl/internet/artykuly/571389,cambridge-analytica-facebook-afera-usa.html> [access: 28.09.2021]

⁸ Z. Wąsik, Z. Kotulski, *Przeptyw informacji w przedsiębiorstwie zarządzanym systemowo* [in:] *Zarządzanie przedsiębiorstwem w warunkach konkurencji*, Wydawnictwo Instytutu Podstawowych Problemów Techniki Polskiej Akademii Nauk, Olsztyn 2002, p. 439.

(which will improve the quality of communication), and also in a spontaneous manner (which will introduce the so-called information noise and which poses a threat to the company related to lowering the effectiveness of its functioning).

5) In the flow of information, information noise should be avoided at all costs, as its negative consequences are manifested primarily in extreme conditions such as breakdowns, additional orders, shortening standard delivery times, new customer requirements combined with short lead times, etc.

6) The area not covered by the systemic management of information flow in the enterprise should be as small as possible.

7) The area covered by the systemic management of information flow in the enterprise should be as large as possible⁹.

2.4. Loss of information in the communication process

When writing about information logistics in an enterprise, it is impossible not to mention the loss of information in the communication process:

- when we speak, we do not pass on about 2 percent of the information we planned to provide (98 percent remain);
- when someone listens to us, they „lose” 1.96 percent of our message on average (96.04 percent remain);
- understanding our speech causes further „losses” – on average 1.92 percent of the message (94.12 percent is left);
- the issue of not accepting some parts of our statement is another „loss” – 1.88 percent (92.4 percent left);
- when the task is performed by only one person, an average of 1.84 percent of the message is captured (90.39 percent is left);

As you can see, the communication process is quite complex and at each of its stages, new „losses” are visible, which make a given task deviate more and more from how it was supposed to be performed in an exemplary manner. In order to minimize these „losses”, it is necessary to use the solutions offered by information logistics¹⁰.

3. Information logistics in e-commerce companies

3.1. The role of information logistics in e-commerce companies

Currently, the overwhelming number of enterprises dealing in the broadly understood trade have expanded their activities or completely transferred them to the Internet, thus joining the dynamically developing group of e-commerce companies around the world. This is a response to the

fact that an increasing percentage of the population has access to the Internet and to the fact that the value of the e-commerce market, both in Poland and abroad is growing all the time¹¹. This was also significantly influenced by the COVID-19 pandemic. The role of information logistics in e-commerce businesses is huge. This is because society in Europe (and not only) has been an information society since the 90s, and thus – eagerly uses modern, developed means of communication and information processing.

3.2. How to take care of information logistics in e-commerce companies?

Since the role of information logistics in e-commerce businesses is crucial, it is necessary to consider how to effectively implement its solutions in the enterprise. Below are five such examples.

1) CRM system – allows you to build relationships between the company and customers, get to know them, gain trust, strengthen relationships.

2) Google Analytics statistics systems – their role is to obtain specific information about potential customers. These are mainly: gender, age, location, interests, searches.

3) Google AdWords and AdSense advertising systems – through them you can find out what type of advertisement is best perceived by users (text advertisement, graphic advertisement, video advertisement), and at what time it should be broadcast to bring the best possible results.

4) Mailing systems – thanks to them, it is possible to examine which topics are of interest to customers and which are not, and which devices are most often used by users. This is the basic information in the process of optimizing the website.

5) Social media – a company that uses social media such as Facebook or Instagram has access to a lot of information about its customers, which can be used in turn by creating advertisements and modernizing the offer¹².

3.3. What are the benefits of using information logistics solutions?

There are many benefits to using information logistics solutions. The most important of them are:

- more effective price management over time;
- more effective use of the advertising budget;
- getting to know customers better, their preferences and expectations;
- the ability to predict customer actions;
- the ability to create a personalized sales offer;
- optimizing the purchasing path;
- reduction of conversion costs;
- improvement of the supply chain management process¹³.

Moreover, the use of information logistics solutions allows enterprises to build a sustainable competitive advantage. This advantage is the source of business success¹⁴.

9. *Ibidem*, p. 7.

10. A. Letkiewicz, A. Mytlewski, *Logistyka informacji w procesach podejmowania decyzji*, „Zeszyty Naukowe Uniwersytetu Gdańskiego” 2005 (30), p. 188.

11. D. Weiland, *Logistyka informacji jako podstawowy element w budowaniu przewagi konkurencyjnej w e-commerce* [in:] „Studia Ekonomiczne. Zeszyty Naukowe Uniwersytetu Ekonomicznego w Katowicach” 2016 (306), p. 104.

12. *Ibidem*, p. 107–108.

13. *Ibidem*, p. 108.

4. Conclusions

Information logistics, used both in everyday life and in professional life, including e-commerce enterprises, is becoming an increasingly important element and there is no sign that this will change. Calling it „the most valuable commodity of the 21st century”, although for some it may be too far-reaching thesis, for more and more people becomes fully justified. Information logistics is very likely to become one of the most studied and used fields of science in the near future¹⁵.



Zezwala się na korzystanie z *Information logistics – the most valuable commodity of the 21st century* na warunkach licencji Creative Commons Uznanie autorstwa 4.0 (znanej również jako CC-BY), dostępnej pod adresem <https://creativecommons.org/licenses/by/4.0/pl/> lub innej wersji językowej tej licencji lub którejkolwiek późniejszej wersji tej licencji, opublikowanej przez organizację Creative Commons.

Bibliography

1. Chaberek M., *Logistyka informacji zarządczej w kontrolingu przedsiębiorstwa*, Wydawnictwo Uniwersytetu Gdańskiego, Gdańsk 2001.
2. *Clickbait – co to jest?*, <https://www.artefakt.pl/blog/epr/clickbait-co-to-jest> [access: 28.09.2021].
3. *Czy powinniśmy skasować Facebooka? Co tak naprawdę zrobiła Cambridge Analytica*, <https://technologia.dziennik.pl/internet/artykuly/571389,cambridge-analytica-facebook-ok-afery-usa.html> [access: 28.09.2021].
4. *Fake news, czyli jak manipulując informacją zdobyć nas w sieci*, <http://fundacjawspomaganiawsi.pl/fake-news-czyli-jak-manipulujac-informacja-zdobyc-nas-w-sieci/> [access: 28.09.2021].
5. Kozłowski R., Sikorski A., *Nowoczesne rozwiązania w logistyce*, Wydawnictwo Wolters Kluwer, Warszawa 2013.
6. Letkiewicz A., Mytlewski A., *Logistyka informacji w procesach podejmowania decyzji*, „Zeszyty Naukowe Uniwersytetu Gdańskiego” 2005 (30), p. 183–198.
7. Wąsik Z., Kotulski Z., *Przepływ informacji w przedsiębiorstwie zarządzanym systemowo* [in:] *Zarządzanie przedsiębiorstwem w warunkach konkurencji*, Wydawnictwo Instytutu Podstawowych Problemów Techniki Polskiej Akademii Nauk, Olsztyn 2002, p. 439–450.
8. Weiland D., *Logistyka informacji jako podstawowy element w budowaniu przewagi konkurencyjnej w e-commerce* [in:] „Studia Ekonomiczne. Zeszyty Naukowe Uniwersytetu Ekonomicznego w Katowicach” 2016 (306), p. 97–110.

14. M. Chaberek, *Logistyka informacji zarządczej w kontrolingu przedsiębiorstwa*, Wydawnictwo Uniwersytetu Gdańskiego, Gdańsk 2001, p. 49.

15. R. Kozłowski, A. Sikorski, *Nowoczesne rozwiązania w logistyce*, Wydawnictwo Wolters Kluwer, Warszawa 2013, p. 81.

Treść roszczenia o usunięcie skutków naruszenia dóbr osobistych w internecie

The content of the claim for the elimination of the consequences of infringement of personal rights on the Internet

Mirosław Dela¹, Piotr Kardasz²

ABSTRACT: The feeling of being anonymous prevalent among Internet users is why personal rights are being infringed very often and with much ease. The outreach of the information published on the Internet brings about far-reaching negative consequences for the injured party, and the elimination of these consequences is extremely difficult and requires the infringing party to submit an appropriate statement, with a big enough outreach so as to effectively reach those recipients who have previously read the content of the defamatory and unlawful entry. The content and the form of such statement are determined by the court in the sentencing part of the judgment. However, before this happens, the claimant must take certain initiative while bringing an action and submit a draft of the statement s/he proposes, together with the manner of its publication. The author of this paper provides guidance on the requirements for publishing such a statement, suggesting what issues should be treated with particular attention during the preparation of the complaint. At the same time, the author does not impose any ready-made solutions, and rather refers to the knowledge in the realm of IT and computer graphics. On the other hand, the author points out to general trends in case law and touches upon the issue of the enforcement of the publication order.

STRESZCZENIE: Poczucie anonimowości użytkowników internetu sprawia, że bardzo często i z wielką łatwością dochodzi do naruszenia dóbr osobistych. Szeroki zasięg informacji publikowanych w internecie niesie dla pokrzywdzonego daleko idące negatywne skutki, których usunięcie jest wyjątkowo trudne i wymaga złożenia przez naruszciciela stosownego oświadczenia, wyeksponowanego na tyle dobrze, by skutecznie dotarło do tego kręgu osób, które zapoznały się uprzednio z treścią oszczerczego i bezprawnego wpisu. Treść i formę tego oświadczenia określa sąd w sentencji wyroku. Zanim to jednak nastąpi, powód musi wykazać się inicjatywą i na etapie składania pozwu wskazać proponowany przez siebie tekst oświadczenia i sposób jego publikacji. Autor niniejszego artykułu udziela wskazówek co do wymogów dotyczących publikacji takiego oświadczenia, sugerując przy tym, na jakie istotne kwestie należy zwrócić szczególną uwagę, sporządzając pozew. Jednocześnie nie narzuca gotowych rozwiązań, odsyłając raczej do wiedzy z zakresu grafiki komputerowej i informatyki. Zwraca za to uwagę na ogólne tendencje panujące w orzecznictwie. Nie pomija również problematyki egzekucji obowiązku publikacji.

KEYWORDS: personal rights, infringement, claim, apology, rectifying

SŁOWA KLUCZOWE: dobra osobiste, naruszenie, roszczenie, przeprosiny, sprostowanie

1. Introduction

The court actions for the protection of personal rights infringed through a publication of a press material (including on the Internet) always involve a conflict between two equivalent goods, i.e. the principle of protecting an individual's personal rights and the right to information (including the freedom of the press)³. Since, as far as the existing law is concerned (*de lege lata*), the provisions of

the press law⁴, do not provide for any special remedies to protect personal rights infringed by a press publication apart from press rectifications⁵. The protection must generally take place in accordance with the provisions of the Act of 23 April 1964 – Polish Civil Code⁶. The prerequisites for the application of remedies under the civil law involve both the infringement of a personal right and the

1. Wrocławska Wyższa Szkoła Informatyki Stosowanej we Wrocławiu, e-mail: mdela@horyzont.eu, ORCID 0000-0002-2078-1049.

2. Wrocławska Wyższa Szkoła Informatyki Stosowanej we Wrocławiu, e-mail: pkardasz@horyzont.eu, ORCID 0000-0002-0933-3384.

3. J. Sadowski, *Postępowanie sądowe w sprawach o ochronę dóbr osobistych za naruszenie tych dóbr spowodowane opublikowaniem materiału prasowego*, LEX/el. 2019, <https://sip.lex.pl/#/publication/470117221> [access: 3.12.2021].

4. The Act of January 26, 1984 – Press Law, consolidated text Journal of Laws of 2018, item 1914.

5. See art. 31a of the cited act.

6. Consolidated text Journal of Laws of 2020, item 1740 as amended. With regard to image protection – also pursuant to art. 81 of the Act of February 4th, 1994 on Copyright and Related Rights, consolidated text Journal of Laws of 2021, item 1062.

unlawfulness of the conduct resulting in such infringement. In art. 24 § 1 of the Polish Civil Code, the legislator indicates, in a very general manner, what may be demanded by the entitled party in the event of the infringement of his/her personal rights, and refers to taking any action necessary to eliminate the consequences of the infringement. This may involve, *inter alia*⁷, „publishing a statement of appropriate content and in an appropriate form”. The enforcement of legal protection requires the person whose personal right has been infringed upon to be an active litigant, which involves not only gathering convincing evidence, but also including a properly formulated claim in their complaint (art. 187 § 1 point 1 and 2 of the Act of 17 November 1964 – the Polish Code of Civil Procedure⁸). Said claim usually takes the form of a request that the defendant should publish the above-mentioned statement. This is intended to compensate for the consequences of the infringement⁹. This statement is the essence of the claim for the elimination of the consequences of an infringement of a personal right. According to the principle of specification, the claimant is responsible for drafting the so-called self-criticism, which the infringing party is then supposed to publish, but it is not enough that the claimant should draft the content of the statement him/herself, but it is also necessary for the claimant to indicate the form in which the statement is to be published¹⁰. The term „form” should be understood as the method of publication, including the layout, or even in broader terms – all IT issues. The lack of specification in this respect is a formal defect of a complaint, which results in the claimant being called to remedy this defect or otherwise the complaint is returned (art. 130 § 1 of the Polish Code of Civil Procedure). It is unacceptable for the claimant to expect that the court would draft and specify the form of the requested statement. The adjudicating body should assess a specific wording and form included in the complaint¹¹. The court

should not substitute the claimant in drafting the statement¹²; the court can, however, interfere with the wording and correct it¹³ if the statement does not serve the purpose or is contrary to the principles of social coexistence¹⁴. It is also unacceptable for the court to say in the sentencing part of the judgment that the wording of the statement is left to be chosen by the defendant infringing party¹⁵.

2. The substance of the statement

The above-mentioned statement usually comes down to a public apology or withdrawal of any allegations formulated (rectification), or to a statement that criticism is unjustified¹⁶. The exact wording of such statement depends in each case on the nature of the right infringed and on the form of the infringement¹⁷. It should be considered as insufficient for the claimant to limit him/herself to a general request that the court should order the defendant to apologize or to withdraw false allegations. The wording of the statement should be adequate to the infringement¹⁸, and at the same time precise, transparent and linguistically correct¹⁹. It should not be worded so as to intentionally humiliate the defendant²⁰. The published statement shall not have a repressive function. Its purpose is only to indicate to the public that certain statements and opinions disseminated by the defendant infringed upon the claimant's personal rights²¹. Certain doubts may arise in a situation when the claimant requests for the publication of a statement that the published information is false or infringing on the claimant's goodwill or privacy. Such a statement may not be made if the issue has not been investigated in the course of the trial²², or if the defendant has not been given the chance to prove the truth²³. The wording of the requested statement should be formulated so that it does not contain any allegations being the original source of the infringement. Literal repetition could

7. The wording of the provision uses the phrase „in particular” denoting an open catalog (*numerus apertus*).

8. Consolidated text Journal of Laws of 2021, item 1805 as amended.

9. P. Machnikowski [in:] E. Gniewek, P. Machnikowski (ed.), *Kodeks cywilny. Komentarz*, Wydawnictwo C.H. Beck, Warszawa 2019, p. 65.

10. Decision of the Supreme Court of 10.10.1989, II CZ 167/89, LEX no. 8991, judgment of the Supreme Court of 13.04.2007, I CSK 28/07, LEX no. 407143, judgment of the Supreme Court of 19.10.2007, II PK 76/07, OSNP 2008/21–22, item 316, judgment of the Supreme Court of 11.02.2010, I CSK 286/09, LEX no. 630167.

11. Judgment of the Supreme Court of 22.12.1997, II CKN 546/97, OSNC 1998/7–8, item 119, judgment of the Supreme Court of 11.02.2010, I CSK 286/09, LEX no. 630167.

12. Judgment of the Court of Appeal in Kraków of 12.01.1994, I ACr 314/93, published in: *Dobra osobiste. Zbiór orzeczeń Sądu Apelacyjnego w Krakowie*, B. Gawlik (ed.), Kraków 1999, p. 100 onwards.

13. P. Machnikowski, *op. cit.* s. 65.

14. E.g. judgment of the Supreme Court of 22.12.1997, II CKN 546/97, OSNC 1998/7–8, item 119, judgment of the Supreme Court of 14.05.2003, I CKN 463/01, OSP 2004/2, item 22, judgment of the Court of Appeal in Warsaw of 26.10.2004, VI ACa 225/04, LEX no. 1642160.

15. Judgment of the Supreme Court of 19.01.1982, IV CR 500/81, OSNC 1982/8–9, item 123, judgment of the Supreme Court of 22.12.1997, II CKN 546/97, OSNC 1998/7–8, item 119, Decision of the Supreme Court of 11.01.2001, V CKN 1852/00, LEX no. 524/02, judgment of the Supreme Court of 14.05.2003, I CKN 463/01, OSP 2004/2, item 22.

16. M. Zaremba, Usunięcie skutków naruszenia dóbr osobistych w internecie – analiza orzecznictwa sądowego, „Przegląd Sądowy” 2020, no. 6, p. 55–68, <https://sip.lex.pl/#/publication/151365494/zaremba-michal-usuniecie-skutkow-naruszenia-dobr-osobistych-w-internecie-analiza-orzecznictwa...?cm=URELATIONS> [access: 3.12.2021].

17. P. Machnikowski, *op. cit.*, p. 65.

18. Judgment of the Supreme Court of 5.06.2009, I CSK 465/08, LEX no. 510611, judgment of the Supreme Court of 19.05.2011, I CSK 497/10, LEX no. 936483.

19. Compare judgment of the Supreme Court of 22.12.1997, II CKN 546/97, OSNC 1998/7–8, item 119, judgment of the Supreme Court of 11.01.2007, II C392/06, LEX no. 276219, judgment of the Supreme Court of 11.02.2010, I CSK 286/09, LEX no. 630167.

20. Judgment of the Court of Appeal in Lublin of 10.07.1998, I ACa 202/98, OSA 2000/2, item 6. See also judgment of the Supreme Court of 13.04.2007, I CSK 28/07, LEX no. 407143.

21. Judgment of the Court of Appeal in Warsaw of 29.11.2018, I ACa 974/17, LEX no. 2691182.

22. Judgment of the Court of Appeal in Warsaw 7.07.2005, I ACa 530/05, LEX no. 1642342.

23. Decision of the Court of Appeal in Kraków of 16.09.1993, I ACr 406/93, B. Gawlik, *op. cit.*, p. 77–84, compare judgment of the Supreme Court of 15.11.2000, III CKN 473/00, LEX no. 51881.

be counterproductive (the so-called boomerang effect²⁴).

3. Audience

As stated above, the provision of art. 24 § 1 of the Polish Civil Code also requires for the claimant to specify the manner of publishing the statement. Like the content, the manner of publication should be adequate to the infringement. In general, this means that the statement should reach the same (or as similar as possible) group of recipients who had previously read the information infringing on the claimant's personal rights²⁵. Only this form of publication can serve the intended purpose of the protective measure undertaken, i.e. to eliminate the consequences of the infringement.

4. Place of publication

In order for the statement to be read by the same group of recipients who have previously read the defamatory information, the form of publication should be similar to the one used with the publication of the defamatory content being the subject of dispute (the so-called „mirror image” principle)²⁶. This requirement may be fulfilled by way of the statement being published in the same place where the defamatory text had been published before²⁷. In practice, this is not always an easy task. The Internet is a specific place for publication. The technical circumstances of the online reality may make proper specification of the form of publishing statements a problematic task. The more precisely this is done, the greater is the risk of a technological complication²⁸. It is often necessary to adapt the form of the statement and the place of its publication to the editorial and technical standards²⁹. The suitability referred to in art. 24 § 1 of the Polish Civil Code means adequacy and proportionality of the form of the statement, which cannot be put on equal terms with mechanical copying of the manner of the defamatory publication³⁰. The specific character of the network environment should be considered³¹. Therefore, it seems that the intention to reach the group of recipients who have previously read (or at least could have read) the defamatory publi-

cation does not necessarily mean that the statement must be published on the same page where the publication violating the claimant's personal rights was published³². In judiciary practice, there is a tendency for the courts adjudicating in cases concerning an infringement of a personal right to order a publication of an apology on the home page of the portal, regardless of where the unlawful content was located in the structure of the website³³. This facilitates the things for the court, and in addition is motivated by the fact that the home page remains the main page, regardless of what changes occur in the structure of the entire portal (subpages) in the future. The statement published on the home page will always remain well displayed, regardless of the changes taking place during its exposure³⁴. In extreme cases, pointing out to a specific subpage in the very judgment (e.g. in the event of a thorough reconstruction of the portal structure), could even result in it being impossible to enforce the judgment. In addition, the content of the apology posted on the home page will grab the attention of those readers who visit the website of this portal and who have read the defamatory publication before. There is a great chance that those readers would not go to the subpage of an article that they had previously read with sufficient attention. This would in fact prevent them from accessing the rectification published later on that same subpage. The statement should be posted on the home page also because of the fact the home page usually contains an announcement of every article, including the one that led to the infringement of personal rights. Thus, the very title (announcement) of the defamatory article (or a link to the subpage with such an article) could depict a negative image of the claimant that, even without it being necessary for the recipients to read the entire article, sufficiently stigmatized him in the eyes of the recipients³⁵. A characteristic feature of the home pages of web portals is frequent fluctuation of the content and the fact that the content rapidly becomes outdated. New announcements are being published frequently, which means that earlier publications are positioned „lower” and are less exposed. Therefore, even good positioning of a statement at the time of publication does not guarantee visibility in the next few hours, and even more so within the next days or weeks. This may also happen by the fault of the defendant himself,

24. Judgment of the Court of Appeal in Kraków of 19.05.1998, I ACa 130/98, B. Gawlik, *op. cit.*, p. 305 onwards. See also judgment of the Supreme Court of 10.05.2007, III CSK 73/07, OSNC-ZD 2008/1, item 28, judgment of the Supreme Court of 2.02.2011, II CSK 431/10, LEX no. 784917. See J. Sadowski, *Naruszenie dóbr osobistych przez media*, Oficyna Naukowa, Warszawa 2003, p. 94–95.

25. M. Zaremba, *op. cit.* Compare judgment of the Supreme Court of 10.09.2009, V CSK 64/09, LEX no. 585910, judgment of the Supreme Court of 8.02.2008, I CSK 345/07, LEX no. 448024, judgment of the Supreme Court of 27.03.2013, I CSK 518/12, OSNC-ZD 2014/1, item 13, judgment of the Supreme Court of 11.08.2016, I CSK 419/15, LEX no. 2087104.

26. Compare e.g. judgment of the Court of Appeal in Warsaw of 12.06.2013, I ACa 35/13, LEX no. 1369391, judgment of the Court of Appeal in Warsaw of 11.02.2014, VI ACa 491/13, LEX no. 1515326, judgment of the Court of Appeal in Warsaw of 6.06.2014, VI ACa 1409/13, LEX no. 1504531, judgment of the Court of Appeal in Warsaw of 19.06.2018, V ACa 520/17, LEX no. 2528159.

27. Compare judgment of the Supreme Court of 19.05.2011, I CSK 497/10, LEX no. 936483.

28. M. Zaremba, *op. cit.*

29. Compare judgment of the Supreme Court of 27.03.2013, I CSK 518/12, unpubl., judgment of the Supreme Court of 5.11.2008, I CSK 164/08, unpubl., judgment of the Supreme Court of 11.04.2006, I CSK 159/05, unpubl.

30. Judgment of the Supreme Court of 5.11.2008, I CSK 164/08, LEX no. 536989, judgment of the Supreme Court of 19.06.2018, V ACa 520/17, LEX no. 2528159.

31. Judgment of the Supreme Court of 18.09.2015, I CSK 813/14, OSNC 2016/7–8, item 93.

32. Judgment of the Court of Appeal in Warsaw of 6.09.2017, VI ACa 636/16, LEX no. 2516046.

33. M. Zaremba, *op. cit.*

34. See judgment of the Court of Appeal in Warsaw of 2.01.2019, VI ACa 1209/17, LEX no. 2706346.

35. Judgment of the Court of Appeal in Warsaw of 2.01.2019 r., VI ACa 1209/17, LEX no. 2706346. Likewise judgment of the Court of Appeal in Warsaw of 23.06.2017, VI ACa 1677/15, LEX no. 2365565.

who usually acts as portal administrator at the same time and has certain decision-making powers as to the structure and the order of information posted there.

Therefore, it is in the claimant's interest that the judgment ordering the publication of the statement requires that it should be maintained on a certain position within the home page and throughout the term indicated in the judgment in order to prevent it from being replaced by other, more current events. Certain websites (e.g. social networking sites) have a functionality of the so-called „pins”, which allows to keep the published post at the top of the page³⁶. In search of the most reasonable measure, one could propose posting the full statement on a portal's subpage (e.g. on the same page where the text violating the claimant's personal rights was posted), and publishing a link to the full text of the statement on the home page. However, this link would have to be clearly visible and, what is more, properly titled, as an average portal reader does not spend too much time reading information which is insufficiently exposed or whose understanding requires above-average intellectual effort. This form of publication, i.e. a clear link to an internal page (subpage), should give an average Internet user a chance to read the content of the statement³⁷. This, in turn, should be considered sufficient, since such a compensatory message is addressed to an average recipient, and not only to those recipients who are particularly interested in the case, especially if the original material was addressed to a „typical” recipient as well. Obviously, this form is not free from the disadvantage of the home page becoming outdated, as indicated above, and this results in less visibility of the referring link. The choice of a given page (subpage) ultimately rests with the court, which does not free the claimant from an obligation to be active while constructing the suggestion contained in the complaint. In addition to selecting the page (or more precisely, the URL) where the statement is supposed to appear, it is also a good thing to determine its location on a given page. Publishing the text of the statement adjacent to other content³⁸, especially among inappropriate entries, may disturb its reception and significantly devalue its meaning. The same may result from allowing other users to add comments under the statement, especially if they are unflattering to the claimant. The claimant may demand that the statement be placed in a specific location of his/her choice (e.g. at the top of the page or under the main news item on the website), and may demand that the court should prohibit the publication of any comments to the statement, advertisements, announcements or other content in its neighborhood, distracting from the essence of the statement³⁹.

5. Term of display of the statement

In the sentencing part of the judgment, the court should set a deadline for the defendant to publish the statement⁴⁰, and specify how long the statement should be visible on the website. Both of these elements are relevant for the elimination of the effects of the infringement, and their detailed specification should be in the claimant's best interest.

Therefore, the requirements with regard to the time of reaction to the violation and with regard to the duration of exposure should constitute an integral part of the claim contained in the complaint. The length of the time when the statement should be kept on the website should depend on at least two factors, i.e.: 1) the time during which the publication containing illegal content was available for viewing (if possible to determine)⁴¹, 2) whether, despite the defendant becoming aware of the infringement of the claimant's personal rights, the defendant has taken any measures to remove the infringing material or limit access thereto⁴². Culpable delay in blocking access to unlawful content should affect the content of the court's judgment even if one were to assume that the judgment in this respect should not perform a repressive function.

6. Ad frame size

In order to eliminate the effects of the infringement, it is imperative that the statement should be legible and clear. This can only be guaranteed by the statement being published in the form of an advertisement of a proper size. Unlike statements published in printed press, websites are displayed on screens with various technical parameters (diagonal and resolution), and most websites automatically adjust to the characteristics of the device where they are displayed⁴³, and this often results in a change in the layout of the elements visible on the website depending on the type of device and the need for the user to scroll the content of the page in the vertical or – even worse – horizontal axis. All of this can negatively affect the reader's perception. Choosing the wrong size, the so-called frame (especially when its width is greater than the width of the screen) may reduce the functionality of the website or may look unnatural, and consequently limit the outreach of the message⁴⁴. One solution could be to define the size of the frame as a fraction of the screen size (e.g. that it should occupy no less than a certain part of it), or by relating the frame size to the size of the font with which the statement text is to be written⁴⁵. The initiative of a claimant having at least basic knowledge in the field of computer graphics, may turn out to be invaluable in this respect.

36. Compare decision of the Court of Appeal in Lublin of 11.10.2018, I ACz 1140/18, unpubl. (issued in the election procedure), decision of the Court of Appeal in Warsaw of 12.09.2019, VI ACz 1143/19, unpubl. (issued in the election procedure).

37. Judgment of the Supreme Court of 19.05.2011, I CSK 497/10, LEX no. 936483.

38. M. Zaremba, *op. cit.*

39. *Ibidem.*

40. It is usually from a few to most often 14 days from the date of issuing the final judgment. *Ibid.*

41. *Ibid.*

42. Judgment of the Court of Appeal in Warsaw of 29.11.2018, I ACa 974/17, LEX no. 2691182.

43. M. Zaremba, *op. cit.*

44. *Ibidem.*

45. *Ibidem.*

7. Font and background parameters

The claim contained in the complaint should also specify the parameters of the font, especially its typeface and size.

The shape of certain fonts makes them illegible or funny-looking and thus not suitable for publishing a statement being an enforcement a court judgment, and issued in the majesty of the law and on behalf of the state. The typeface may be specified by indicating a specific font name or by referring to the typeface used in other texts on the page.

The latter solution promotes harmony and makes the receipt of message more pleasing. The claimant's requests may also concern more detailed issues, e.g. bolding, italics, underlining, justification, the use of capital letters, line spacing, the width of the margin between the text and the frame edge⁴⁶, as well as the font color etc.

The latter, especially when in juxtaposition with the color of the background against which the statement is to be published, is of great importance when it comes to the contrast and legibility of its content. Improper selection of the color palette, or leaving this aspect to the discretion of the defendant, may result in the publication of the statement in a color that will make it difficult to read.

The font size should not differ significantly from the font size used in other texts on the same page, and it certainly cannot be significantly smaller⁴⁷. Sometimes the courts define the required size as „standard and readable”, which is not a sufficiently precise formulation⁴⁸.

Equally insufficient is requesting that the font size is supposed to make the statement readable by the average user⁴⁹.

A mistake is not providing any precise guidelines and leaving any disputes in this regard to be resolved during the enforcement of the judgment.

The court's failure to specify the font size of the statement may persuade the defendant to publish the statement in a font so small that reading its content will be significantly difficult. The font size can be specified in absolute terms (in pixels or typographic points)⁵⁰ or relative, referring to the frame size⁵¹ or to the font size in articles adjacent to the statement⁵².

It is also possible to use a font of the size and typeface used in the defamatory publication.

46. *Ibidem*.

47. *Ibidem*.

48. Compare judgment of the District Court of 17.03.2017, IV C 997/15, LEX no. 2300672.

49. See judgment of the Court of Appeal in Warsaw of 23.06.2017, VI ACa 1677/15, LEX no. 2365565.

50. M. Zaremba, *op. cit.*

51. Compare e.g. judgment of the Court of Appeal in Warsaw of 28.03.2018, VI ACa 1751/16, LEX no. 2545154, judgment of the District Court in Warsaw of 9.06.2017, I C 125/16, unpubl.

52. Compare judgment of the Court of Appeal in Warsaw of 14.04.2016, VI ACa 1847/14, LEX no. 2075684.

53. M. Zaremba, *op. cit.*

54. See judgment of the Court of Appeal in Kraków of 19.09.2012, I ACa 703/12, LEX no. 1236715, judgment of the Court of Appeal in Warsaw of 9.12.2015, VI ACa 1772/14, LEX no. 2009536.

55. Compare judgment of the District Court in Lublin of 11.06.2014, I C 23/11, LEX no. 1845367.

56. J. Sadomski, *Postępowanie sądowe...*, LEX/el. 2019.

57. Judgment of the Court of Appeal in Warsaw of 12.12.2012, VI ACa 259/12, LEX no. 1362958.

58. Judgment of the Supreme Court of 22.12.1997, II CKN 546/97, judgment of the Supreme Court of 14.05.2003, I CKN 463/01, OSP 2004/2, item 22.

59. J. Sadomski, *Postępowanie sądowe...*, *op. cit.* Judgment of the Supreme Court of 9.07.1971, II CR 220/71, OSNC 1972/1, item 19, judgment of the Supreme Court of 19.01.1982, IV CR 500/81, OSNC 1982/8–9, item 123, judgment of the Supreme Court z 5.06.2009, I CSK 465/08.

8. Counteracting the manipulation of the infringing party

As there is usually a subjective identity between the defendant infringing party and the administrator of the website, it should be expected to take actions aimed at making it difficult for the reader to read the statement. Information technology offers countless possibilities for manipulating the content of websites. The claimant may demand that the court should order the publication of the statement so that every user can read it without taking any additional steps⁵³, e.g. without the need to install additional software. Also, the software installed on the user's device (e.g. blocking unwanted advertisements) should not prevent the visibility of the statement⁵⁴. Courts, for understandable reasons, avoid delving into technical details, which is why the judgments contain a general requirement that any measures that diminish the meaning and gravity of the text should not be used⁵⁵. However, there is nothing to prevent the claimant from demanding that the content of this general prohibition be clarified.

9. Granting legal protection

The form and the content of non-financial remedies must be adequate to the form and the content of the infringement and necessary to eliminate its effects. The principles of proportionality⁵⁶ and purposefulness should apply in this respect. Although the court is not strictly bound by the claimant's claim indicated in the complaint in terms of the text or the form of publication, the court should take those factors into account, if the claim is reasonable enough⁵⁷.

Under art. 24 § 1 of the Polish Civil Code, it is possible to infer the court's obligation to control both the content and the form of the requested statement, whether they are appropriate and necessary to eliminate the consequences of the infringement.

The court may interfere with the requested text of the statement by limiting its scope or detailing the wording thereof⁵⁸. The judgment may not be limited to the general prohibition of infringement of the claimant's personal rights⁵⁹. When deciding on the obligation, the court should *ex officio* take into account the costs of publishing the statement, making sure that the implementation

of the legitimate non-pecuniary right of the injured party does not lead to an excessive and unreasonable financial burden on the infringing party⁶⁰. The proportionality of the applied remedies is important because any abuse in that regard discourage journalists from taking up controversial topics, even though they are important from the point of view of public debate (the so-called chilling effect)⁶¹. The request to eliminate the consequences of infringement of the claimant's personal rights by publishing an appropriate statement may be accompanied by an additional claim for the application of property protection measures, i.e. the payment of monetary compensation or the payment of an appropriate amount of money for a specific public purpose (art. 448 of the Polish Civil Code in conjunction with art. 24 § 1 of the Polish Civil Code). If a damage to property was caused as a result of an infringement of a personal right, the injured party may demand compensation (damages) on general terms (art. 415 of the Polish Civil Code in conjunction with art. 24 § 2 of the Polish Civil Code).

10. Enforcement of the obligation to publish a declaration

The obligation to eliminate the effects of infringement of the claimant's personal rights, meaning that the infringing party is ordered to a statement of appropriate content and in an appropriate form, constitutes the so-called an interchangeable activity that may be subject to a specific type of enforcement by way of substitute performance (art. 1049 § 1 of the Polish Code of Civil Procedure). Anticipating a possible delay on the part of the debtor in performance of the obligation, the claimant may, pursuant to art. 480 § 1 of the Polish Civil Code and in the very complaint, submit a request for authorization for the claimant to publish the statement him/herself, with the content indicated in the judgment and at the expense of the defendant⁶², in case the obligation imposed upon the defendant is not fulfilled⁶³, or in case the fulfillment of the defendant's obligation is improper⁶⁴. It cannot be ruled out that there will be a hypothetical situation where the enforcement of the judgment may prove impossible. This will happen when the ordered form of publishing the statement does not take into account the technological conditions of the Internet, or when the obligation is formulated in too vague words, or is too detailed and inflexible. In such circumstances, the defendant will be released from the obligation imposed on him/her in accordance with the Latin principle of *impossibilium nulla obligatio est*. It is in the claimant's interest that the judgment

should be formulated in a sufficiently precise and flexible manner to enable the fulfillment of the obligation in any circumstances. However, it should not be the claimant's intention to impose an obligation that would be too onerous for the defendant⁶⁵, because in such circumstances, the defendant could deliberately hinder and delay the performance of the obligation.

11. Summary

When resolving the conflict between the principle of freedom of speech and the principle of protection of the an individual's personal rights in favor of the latter, and thus granting protection to personal rights infringed upon by a press publication, the court imposes an obligation on the defendant to submit an appropriate statement. In order for the judgment to be enforceable and its enforcement not to cause additional problems, it is necessary to provide details of both the substance of the statement and the form in which it is to be published in a sufficiently precise and flexible manner. In order to be adequate, the specification of the method for publishing the requested statement must take into account the technological conditions of the communication technique used, and in addition, must make sure that improperly excessive form does not undermine its substance. Contrary to what may seem, publishing a statement on the Internet may cause more problems than publication in any other medium. It requires the adjudicating panel to know at least the basic issues in the field of computer graphics and the functioning of websites, and in extremely complicated cases to consult an expert with special knowledge (art. 278 § 1 of the Polish Code of Civil Procedure). In this matter, the claimant may significantly facilitate the task for the court if the complaint correctly specifies both the content and the form of the requested statement.

Bibliography

1. Gawlik B., *Dobra osobiste. Zbiór orzeczeń Sądu Apelacyjnego w Krakowie*, Kantor Wydawniczy Zakamycze, Kraków 1999.
2. Gniewek E., Machnikowski P. (ed.), *Kodeks cywilny. Komentarz*, Wydawnictwo C.H. Beck, Warszawa 2019.
3. Sadowski J., *Naruszenie dóbr osobistych przez media. Analiza praktyki sądowej*, Oficyna Naukowa, Warszawa

60. P. Machnikowski, *op. cit.*, p. 65, judgment of the Supreme Court of 17.05.2013, I CSK 540/12, Legalis, judgment of the Supreme Court of 18.09.2015, I CSK 813/14 Legalis.

61. See also judgement of European Court of Human Rights of 7.06.2007, Dupuis et al. v. France, application no.1914/02, judgement of European Court of Human Rights of 23.04.2015, Morice v. France, application no. 29369/10.

62. At the request of the creditor, the court will grant him the sum needed to perform the activities (art. 1049 § 1 Polish Code of Civil Procedure).

63. Resolution of the Supreme Court of 28 Jun 2006, III CZP 23/06, OSNC 2007/1, item 11. Compare also judgment of the Supreme Court z 23 May 2013, I CSK 531/12, LEX no. 1383225, resolution of the Supreme Court of 28 Jun 2008, III CZP 23/06, OSNC 2007/1, item 11, resolution of the Supreme Court of 17 Feb 2016, III CZP 106/15, OSNC 2017/2, item 13.

64. Judgment of the District Court in Słupsk of 17 Nov 2016, I C 266/16, LEX no. 2162200.

65. M. Zaremba, *op. cit.*

2003.

4. Sadowski J., *Postępowanie sądowe w sprawach o ochronę dóbr osobistych za naruszenie tych dóbr spowodowane opublikowaniem materiału prasowego*, LEX/el. 2019.

5. Zaremba M., *Usunięcie skutków naruszenia dóbr osobistych w internecie – analiza orzecznictwa sądowego*, „Przegląd Sądowy” 2020, no. 6.

6. The Act of April 23, 1964 – Polish Civil Code (consolidated text Journal of Laws of 2020, item 1740 as amended).

7. The Act of November 17, 1964 – the Polish Code of Civil Procedure (consolidated text Journal of Laws of 2021, item 1805 as amended).

8. The Act of January 26, 1984 – Press Law (consolidated text Journal of Laws of 2018, item 1914).

9. The Act of February 4th, 1994 on Copyright and Related Rights (consolidated text Journal of Laws of 2021, item 1062).

10. Internetowy System Aktów Prawnych <https://isap.sejm.gov.pl/>

11. System Informacji Prawnej LEX <https://www.lex.pl/>



Zezwala się na korzystanie z *The content of the claim for the elimination of the consequences of infringement of personal rights on the Internet* na warunkach licencji Creative Commons Uznanie autorstwa 4.0 (znanej również jako CC-BY), dostępnej pod adresem <https://creativecommons.org/licenses/by/4.0/pl/> lub innej wersji językowej tej licencji lub którejkolwiek późniejszej wersji tej licencji, opublikowanej przez organizację Creative Commons.

Analiza i implementacja algorytmów odnajdywania ścieżki do zastosowania w grach przeglądarkowych

Analysis and implementation of pathfinding algorithms for use in browser games

Beata Laszkiewicz¹, Tomasz Sobczak²

STRESZCZENIE: Celem tego artykułu jest przedstawienie, porównanie oraz implementacja algorytmów odnajdywania ścieżki do zastosowania w grach przeglądarkowych z wykorzystaniem ogólnodostępnych, darmowych technologii internetowych. Pokazano również możliwość wykorzystania najlepszego algorytmu w grze przeglądarkowej.

ABSTRACT: The goal of this article is to present, compare and implement path finding algorithms for use in browser games, using public, free internet technologies. The possibility of using the best algorithm in a browser game is also shown.

SŁOWA KLUCZOWE: gra przeglądarkowa, graf, algorytm Bellmana-Forda, algorytm Dijkstry, algorytm A-star

KEY WORDS: browser game, graph, Bellman-Ford algorithm, Dijkstra algorithm, A-star algorithm

1. Wprowadzenie

Ciągły rozwój technologii sprawił, że niegdyś proste i statyczne strony internetowe, które nie pozwalały na żadną interakcję z użytkownikiem, dziś mogą reagować nawet na najmniejszy gest. W dobie Internetu, kiedy ogromną popularnością cieszą się portale i serwisy społecznościowe, swoje miejsce znalazł również przemysł gier. Dzięki takim projektom jak Facebook czy Google+ producenci uzyskali dynamicznie rosnący rynek zbytu – ogromną rzeszę osób, która w innym przypadku nie zainteresowałaby się ich produktami. Jednak gdy te stały się elementem ulubionych serwisów społecznościowych, zaczęły być odbierane jako coś wartościowego, w co warto spróbować zagrać.

Gry przeglądarkowe mogą być bardziej lub mniej zaawansowane, jednakże niezależnie od tego, w jakiej technologii zostały stworzone, bardzo często wykorzystują podobne algorytmy i wzorce projektowe. Jednym z typów takich właśnie algorytmów są algorytmy wyszukiwania ścieżki. Algorytmy odnajdywania ścieżek są ściśle powiązane z zagadnieniami sztucznej inteligencji oraz teorii grafów. Pierwotnie były wykorzystywane do odnajdywania najkrótszych odległości pomiędzy wierzchołkami grafu, ale wraz z rozwojem technologii i sztucznej inteligencji zaczęto ich używać m.in. przy wyszukiwaniu i planowaniu tras.

2. Podstawy teoretyczne

Teoria grafów to obszerny dział matematyki i informatyki, zajmujący się badaniem właściwości grafów. Grafy możemy podzielić na proste (grafy nieskierowane, grafy), skierowane (digrafy) oraz mieszane.

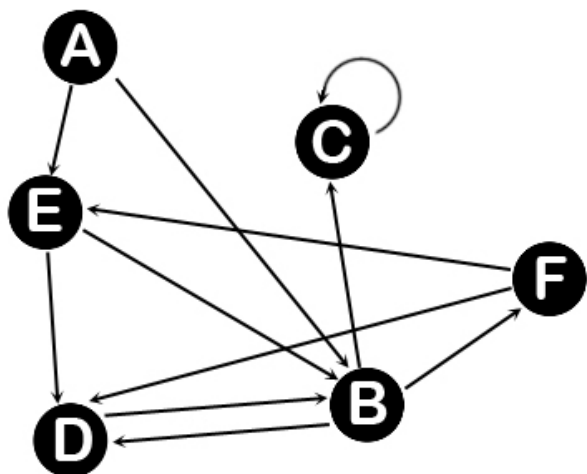
Graf to zbiór wierzchołków (węzłów), które mogą być połączone krawędziami w taki sposób, że każda krawędź kończy się i zaczyna w którymś z wierzchołków. Dodatkowo krawędzie mogą posiadać swoją wagę (koszt przejścia), którą najczęściej można wyobrazić sobie jako odległość lub czas przejścia pomiędzy wierzchołkami, jednakże ich interpretacje są nieograniczone. Dla każdego węzła możemy również określić jego stopień, czyli liczbę krawędzi incydentnych z tym węzłem. Graf skierowany charakteryzuje się krawędziami, dla których określony jest jeden kierunek. Takie krawędzie pozwalają przejść z jednego wierzchołka grafu do drugiego, ale nie na odwrót. Grafy mieszane jednocześnie posiadają zarówno krawędzie nieskierowane, jak i skierowane³.

Na rys. 1 oraz rys. 2 przedstawiono przykłady grafów: skierowanego i nieskierowanego.

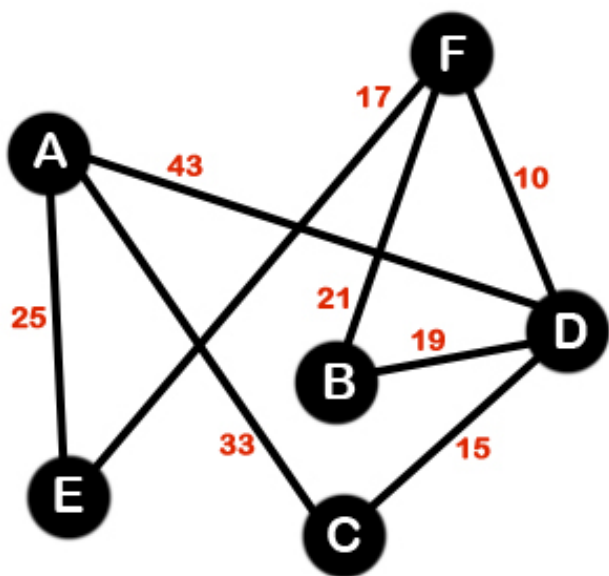
1. Wrocławska Wyższa Szkoła Informatyki Stosowanej we Wrocławiu, e-mail: blaszkiewicz@horyzont.eu, ORCID 0000-0002-1447-4755.

2. Wrocławska Wyższa Szkoła Informatyki Stosowanej we Wrocławiu, e-mail: tomasz-chudzik@wp.pl.

3. K. A. Ross, C. R. B. Wright, *Matematyka dyskretna*, Wydawnictwo Naukowe PWN, Warszawa 1996, s. 327–336. M. M. Sysło, N. Deo, J. S. Kowalik, *Algorytmy optymalizacji dyskretnych*, Wydawnictwo Naukowe PWN, Warszawa 1995, s. 179.

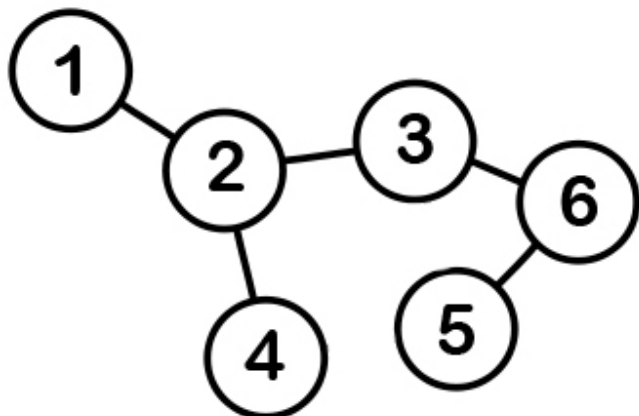


Rys. 1. Przykład grafu skierowanego⁴

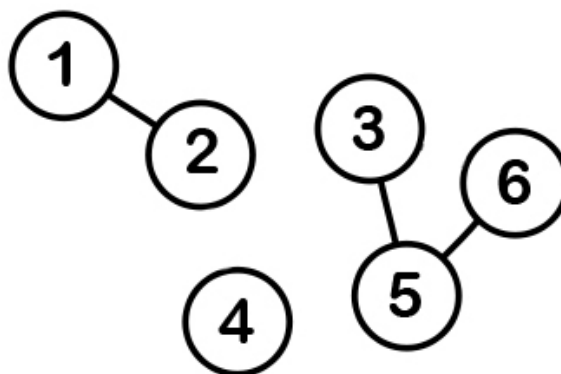


Rys. 2. Przykład grafu nieskierowanego wraz z wagami

Grafy można również podzielić ze względu na różne właściwości – tzw. klasy. Najczęściej spotykane z nich to grafy spójne, grafy acykliczne oraz drzewa. Z grafem spójnym mamy do czynienia, gdy między jego dowolnymi dwoma węzłami istnieje łącząca je ścieżka, czyli z dowolnego wierzchołka grafu można dotrzeć do każdego innego. Przykłady grafu spójnego i niespójnego znajdują się na rys. 3. i rys. 4.

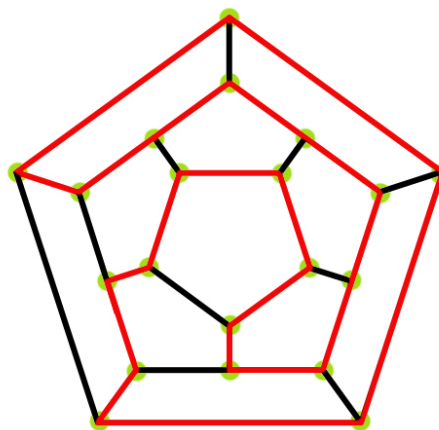


Rys. 3. Przykład grafu spójnego

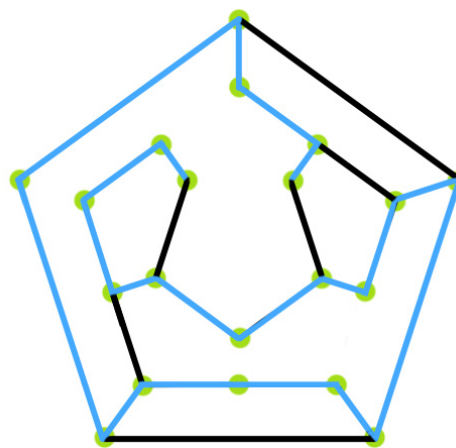


Rys. 4. Przykład grafu niespójnego

Grafy acykliczne nie posiadają cykli, czyli ścieżek, których koniec pokrywa się początkiem, i nie zawierają innych powtarzających się węzłów lub krawędzi. Na rysunku 5. został przedstawiony graf z oznaczonym cyklem Hamiltona, czyli cyklem, którego ścieżka nie przechodzi przez żaden z wierzchołków dwukrotnie. Na rys. 6. zaprezentowano jego podgraf (graf powstały przez usunięcie z grafu nadrzędnego pewnej liczby wierzchołków lub krawędzi) tworzący graf acykliczny. Na niebiesko zaznaczono ścieżkę przechodzącą przez wszystkie wierzchołki, lecz nietworzącą cyklu (ścieżka nie została zamknięta).



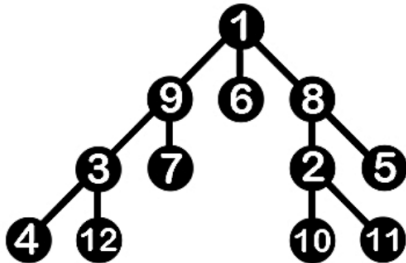
Rys. 5. Przykład grafu cyklicznego



Rys. 6. Przykład grafu acyklicznego

⁴ Wszystkie rysunki są autorstwa T. Sobczaka.

Drzewami nazywamy grafy, które są jednocześnie acykliczne oraz spójne. Drzewo, w którym jeden z wierzchołków został wyróżniony, nazywane jest drzewem ukorzenionym, a wyróżniony wierzchołek – korzeniem. Na rys. 7 został przedstawiony przykład drzewa ukorzonego, którego korzeniem jest wierzchołek oznaczony cyfrą 1.



Rys. 7. Przykład drzewa ukorzonego

2.1. Algorytm Bellmana-Forda

Algorytm ten wykorzystywany jest najczęściej do odnajdywania optymalnej trasy w sytuacjach, gdy nie liczy się czas, a wynik. Pierwotnie służył do odnajdywania ścieżek o najmniejszej wadze pomiędzy wszystkimi wierzchołkami danego grafu.

Algorytm opiera się na metodzie relaksacji, czyli sprawdzeniu, czy przy przejściu daną krawędzią grafu nie otrzymamy trasy krótszej (mniej kosztownej) niż dotychczas znana ścieżka. Niestety algorytm ze względu na swoją budowę posiada dużą złożoność czasową – $O(|V| \cdot |E|)$, gdzie V to zbiór wierzchołków, a E to zbiór krawędzi między wierzchołkami. Zaletą tego algorytmu jest możliwość użycia go dla grafu zawierającego ujemne wagi przejść między wierzchołkami⁵. Pełny opis algorytmu można znaleźć w pracy *Analiza i implementacja algorytmów odnajdywania ścieżki do zastosowania w grach przeglądarkowych*⁶.

2.2. Algorytm Dijkstry

W przeciwieństwie do algorytmu Bellmana-Forda ten algorytm nie zezwala na ujemne koszty krawędzi między węzłami. Algorytm Dijkstry zachowuje się podobnie jak algorytmy zachłanne, ponieważ wybiera zawsze pole o najmniejszym koszcie (i nie powraca już do niego). Dzięki tym założeniom złożoność czasowa algorytmu jest dużo mniejsza: przy zastosowaniu do przechowywania informacji kopca binarnego, czyli struktury danych reprezentującej drzewo, w którym stopień każdego wierzchołka nie może być większy od 3, wynosi $O(|E| \log |V|)$, gdzie V to zbiór wierzchołków, a E to zbiór wszystkich krawędzi

pomiędzy wierzchołkami⁷. Pseudokod oraz omówienie algorytmu można znaleźć w pracy *Analiza i implementacja [...]*⁸.

2.3. Algorytm A*

Algorytm A* (A-Star) jest rozwinięciem algorytmu Dijkstry. Dzięki zastosowaniu heurystyki (szacowania) skrypt jest w stanie wyznaczyć optymalną trasę stosunkowo niewielkim kosztem czasowym. Przy zastosowaniu odpowiedniej heurystyki algorytm może osiągnąć wyniki w czasie istotnie mniejszym niż algorytm Dijkstry.

Algorytm przeszukuje graf (mapę) poprzez wędrówkę po ścieżce, która według obecnie znanych kosztów jest najniższa, jednocześnie zapamiętując alternatywne rozwiązania. Jeżeli jakiś element trasy osiągnie koszt wyższy niż alternatywne rozwiązanie, wtedy obecny kierunek jest porzucany na koszt tańszego rozwiązania. Proces ten trwa, dopóki cel nie zostanie osiągnięty.

W tabeli 1. przedstawiono dwie najczęściej wykorzystywane funkcje heurystyczne, które oszacowują odległość pomiędzy dwoma punktami. Zmienna A to punkt początkowy, zmienna B to punkt końcowy, a indeksy przy zmiennych oznaczają współrzędne punktu, czyli oznaczenie A_x to współrzędna x dla punktu A.

| Nazwa funkcji | Wzór |
|---------------------|--|
| Manhattan | $H = B_x - A_x + B_y - A_y $ |
| Odległość Euklidesa | $H = \sqrt{(B_x - A_x)^2 + (B_y - A_y)^2}$ |

Tab. 1. Popularne funkcje heurystyczne

Pełny pseudokod algorytmu oraz jego szczegółowy opis można znaleźć w pracy *Analiza i implementacja [...]*⁹.

3. Środowisko programistyczne

Dostępnych jest wiele języków programowania, które mogą zostać wykorzystane do implementacji przedstawionych wcześniej algorytmów, ale pod uwagę wzięto jedynie te darmowe i cieszące się w ostatnich czasach popularnością. Jeśli chodzi o sposób przechowywania danych, zastosowano rozwiązania korzystające z języka SQL (m.in. MySQL) oraz takie, które kierują się obecnie modną zasadą NoSQL (m.in. MongoDB).

Jako język programowania, w którym zaimplementowane zostaną przedstawione wcześniej algorytmy, wybrano JavaScript ze względu na jego uniwersalność. Jego zaletami są obsługa przez wszystkie obecne przeglądarki internetowe oraz możliwość użycia jednego języka programowania do stworzenia frontendu i backendu aplikacji.

Do przechowywania danych na serwerze wykorzystany

5. *Algorytmy i struktury danych*, wykład 7: http://hector.tu.kielce.pl/przedmioty/aisd-wyk/AiSD_w07.pdf, [dostęp: 30.11.2021].

6. T. Sobczak, *Analiza i implementacja algorytmów odnajdywania ścieżki do zastosowania w grach przeglądarkowych*, inżynierska praca dyplomowa, Wrocławska Wyższa Szkoła Informatyki Stosowanej, Wrocław 2012.

7. K. A. Ross, C. R. B. Wright, *op. cit.*, M. M. Sysło, N. Deo, J. S. Kowalik, *op. cit. Algorytmy i struktury danych, op. cit.*

8. T. Sobczak, *op. cit.*

9. *Ibidem*.

zostanie format JSON, gdyż dane nie muszą być przechowywane w sposób relacyjny i nie będą potrzebne kwerendy czy innego typu zapytania. Ponadto cechuje go prostota przekazania informacji do skryptu JavaScript – dzięki zastosowaniu takiego rozwiązania wszelkie informacje (np. o terenie) zostaną dostarczone do użytkownika aplikacji bardzo szybko.

4. Implementacja algorytmów

4.1. Algorytm Bellmana-Forda

W listingu 1. znajduje się główna pętla algorytmu, która jest odpowiedzialna za relaksację krawędzi. Przed rozpoczęciem pętli miało miejsce przygotowanie zmiennych. W przypadku, gdy nie ma już żadnych krawędzi, które mogłyby zostać zrelaksowane, pętla zostaje przerwana.

Zmienne wykorzystane w funkcji:

Vertex – zbiór wszystkich pól na mapie

Edges – zbiór krawędzi pomiędzy tymi polami

```

1. for (i in Vertex) {
2.   r = false;
3.   for (j in Edges) {
4.     e = Edges[j];
5.     if (Vertex[e.v].cost > Vertex[e.u].cost + e.w) {
6.       Vertex[e.v].cost = Vertex[e.u].cost + e.w;
7.       Vertex[e.v].parent = e.u;
8.       r = true;
9.     }
10.  }
11.  if (!r) {
12.    break;
13.  }
14. }
```

Listing 1. Implementacja algorytmu Bellmana-Forda¹⁰

4.2. Algorytm Dijkstry

Algorytm Dijkstry wykorzystuje kopce binarne do reprezentacji zbiorów, co wpływa na zwiększenie wydajności. Algorytm działa do momentu, gdy aktualny węzeł trafi do zbioru S lub zbiór Q będzie pusty – to kolejny krok w kierunku optymalizacji.

Zmienne wykorzystane w skrypcie (zob. listing 2.):

Q – zbiór odwiedzonych węzłów

S – zbiór węzłów, które algorytm uznał za całkowicie sprawdzone

P – tablica relacji zachodzących pomiędzy węzłami (połączenia typu rodzic-dziecko)

M – tablica informacji na temat pola

```

4. y = M[m.id].y;
5. S[m.id] = true;
6. if (x == end_x && y == end_y) {
7.   break;
8. }
9. for (i = 1; i <= 6; ++i) {
10.  switch (i) {
11.   case 1:
12.     k = x;
13.     l = y - 1;
14.     break;
15.   case 2:
16.     k = x + 1;
17.     l = y;
18.     break;
19.   case 3:
20.     k = x + 1;
21.     l = y + 1;
22.     break;
23.   case 4:
24.     k = x;
25.     l = y + 1;
26.     break;
27.   case 5:
28.     k = x - 1;
29.     l = y;
30.     break;
31.   case 6:
32.     k = x - 1;
33.     l = y - 1;
34.     break;
35.  }
36.  if (accessibility(k,l)) {
37.    n = map[k][l];
38.    if (!S[n.id]) {
39.      z = Q.info(n.id);
40.      if ( z ) {
41.        if (m.cost + n.cost < z.cost) {
42.          Q.remove(n.id);
43.          Q.push({ id : n.id, cost : m.cost + n.cost});
44.          P[n.id] = m.id;
45.        }
46.      } else {
47.        Q.push({ id : n.id, cost : m.cost + n.cost});
48.        P[n.id] = m.id;
49.      }
50.    }
51.  }
52. }
53. }
```

Listing 2. Implementacja algorytmu Dijkstry

4.3. Algorytm A*

Algorytm A*, jak wspomniano wcześniej, jest bardzo podobny do algorytmu Dijkstry, dlatego też w listingu 3.

```

1. while (Q.size() > 0) {
2.   m = Q.pop();
3.   x = M[m.id].x;
```

10. Wszystkie listingi są autorstwa T. Sobczaka.

przedstawiono tylko fragment zaczynający się od linii 36 – cały poprzedzający ją kod jest identyczny jak w algorytmie Dijkstry.

Wykorzystane zmienne również są podobne:

- Q – zbiór węzłów otwartych (odwiedzonych)
- S – lista węzłów zamkniętych (uznanych za sprawdzone)
- P – tablica relacji zachodzących między wierzchołkami

```

36. if (accessibility(k,l)) {
37.   n = map[k][l];
38.   if (!S[n.id]) {
39.     z = Q.info(n.id);
40.     if ( z ) {
41.       g = m.g + M[n.id].cost;
42.       if (g < z.g) {
43.         h = heuristic(k, l, end_x, end_y);
44.         Q.remove(n.id);
45.         Q.push({ id : n.id, f : g + h, g : g, h : h });
46.         P[n.id] = m.id;
47.       }
48.     } else {
49.       h = heuristic(k, l, end_x, end_y);
50.       g = m.g + M[n.id].cost;
51.       Q.push({ id : n.id, f : g + h, g : g, h : h });
52.       P[n.id] = m.id;
53.     }
54.   }
55. }
56. }
57. }

```

Listing 3. Implementacja algorytmu A*

W listingu 4. przedstawiono funkcję heurystyczną bazującą na metodzie Manhattan oraz niedoszacowanym modelu heurystyki. Dzięki niedoszacowaniu funkcja zawsze zwróci najniższy możliwy koszt ruchu, przez co algorytm znajdzie najlepszą ścieżkę.

Na wejściu funkcja pobiera współrzędne pola startowego (x_1 i y_1) oraz pola końcowego (x_2 i y_2). Na wyjściu funkcja zwraca szacowany koszt potrzebny do przejścia od punktu startowego do końcowego.

```

1. function heuristic (x1, y1, x2, y2) {
2.   var distance, abs = Math.abs, max = Math.max,
3.   dx = x2 - x1,
4.   dy = y2 - y1,
5.   dd = dy - dx;
6.   distance = max(abs(dd), max(abs(dx), abs(dy)));
7.   return distance * min_koszt;
8. }

```

Listing 4. Implementacja funkcji heurystycznej Manhattan

Funkcja przeszacowana różni się od niedoszacowanej jedynie wartością, przez którą mnoży się dystans – zamiast najniższego występującego na mapie kosztu ruchu (najniż-

szej wagi krawędzi w grafie) użyty jest koszt najwyższy:

```
7. return distance * max_koszt;
```

Listing 5. Zastosowanie najwyższego kosztu w funkcji heurystycznej Manhattan

Przeszacowanie znacznie skraca czas obliczeń (algorytm najczęściej wybiera pierwsze pole, na które trafi, jako to najtańsze). Odbija się to jednak na dokładności – bardzo często będzie istnieć przynajmniej jedna ścieżka, której koszt okaże się niższy niż odnaleziony przez algorytm z taką heurystyką.

4.4. Elementy wspólne

Poniżej opisano elementy używane przez każdy z algorytmów. Pierwsza funkcja (listing 6.) odpowiada za sprawdzenie dostępności pola. Za parametry wejściowe przyjmuje współrzędne x i y pola, które ma zostać sprawdzone. Na wyjściu funkcja zwraca wartość logiczną (`true` lub `false`) oznaczającą dostępność pola. Pole jest dostępne, jeżeli istnieje na mapie oraz koszt wejścia na to pole jest mniejszy niż 999 punktów (wartość symboliczna przyjęta przez Autorów pracy). Funkcja wykorzystuje również zmienną globalną `worldMap`, w której przechowywane są informacje o planszy.

```

1. function accessibility (x, y) {
2.   var map = worldMap;
3.   if (typeof map[x] === 'undefined'){
4.     return false;
5.   }
6.   if (typeof map[x][y] === 'undefined'){
7.     return false;
8.   }
9.   if (map[x][y].cost >= 999){
10.    return false;
11.  }
12.  return true;
13. }

```

Listing 6. Funkcja accessibility

Funkcja przedstawiona na listingu 7. jest odpowiedzialna za konwersję wyniku zwróconego przez algorytm i podanie poprawnej ścieżki (zmienna `path`). W tym celu funkcja pobiera identyfikator pola startowego (`start`) i końcowego (`stop`), identyfikator następnego pola (`next`) oraz tablicę relacji pomiędzy polami (`P`). Zmienna `path` odpowiada za przechowywanie ścieżki. Algorytm wykonuje się rekurencyjnie, krok po kroku uzupełniając zmienną `path` o kolejne węzły. Funkcja przerywa swoje działanie, gdy aktualny węzeł jest polem końcowym.

```

1. function parsePath (start, stop, next, P, path) {
2.   path.push(next);
3.   if (next === stop) {

```

```

4. return path;
5. } else {
6. next = P[next];
7. return parsePath(start, stop, next, P, path);
8. }
9. }

```

Listing 7. Funkcja parsePath

W listingu 8. przedstawiono blok kodu odpowiedzialny za przygotowanie ścieżki do graficznego przedstawienia jej na mapie. Na początku deklarowane są zmienne, które zostaną przekazane do opisanej wcześniej funkcji `parsePath`, oraz zmienne pomocnicze dla pętli. Warto zauważyć, że funkcja `parsePath` wędruje od pola końcowego do początkowego – takie działanie jest wymuszone sposobem tworzenia relacji pomiędzy węzłami. W każdym kroku pętli do zmiennej `path` dodawane są wartości `x`, `y` oraz `cost`, które zostaną potem wykorzystane przy renderowaniu ścieżki w module gry.

```

1. var start_id = map[end_x][end_y].id,
2. stop_id = map[start_x][start_y].id,
3. temp = parsePath(start_id, stop_id, start_id, P, []),
4. path = {}, counter;
5.
6. for (counter = 0; counter < temp.length; counter++) {
7. i = temp[counter];
8. if (typeof M[i] !== 'undefined') {
9. path[(counter+1)] = [];
10. path[(counter+1)][1] = M[i].x;
11. path[(counter+1)][2] = M[i].y;
12. path[(counter+1)][3] = M[i].cost;
13. }
14. }

```

Listing 8. Blok przygotowujący ścieżkę do przedstawienia na mapie

5. Analiza algorytmów

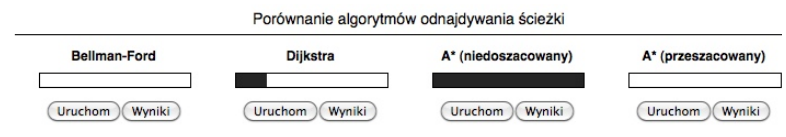
W dzisiejszych czasach różnicowanie sprzętu komputerowego jest duże, dlatego też testy zostały przeprowadzone w szerszej grupie kontrolnej: w pięćdziesięciu różnych środowiskach sprzętowych, w których każdy z testów powtórzono dwudziestokrotnie.

5.1. Aplikacja do testów

Specjalnie na potrzeby testów została napisana aplikacja umożliwiająca w bardzo prosty sposób przeprowadzenie badań przez osoby trzecie. Na rys. 8. przedstawiono jej interfejs.

Wszystkie algorytmy miały zakodowany punkt startowy oraz końcowy wyszukiwanej trasy, dzięki czemu każdorazowe uruchomienie funkcji implementującej dany algorytm odbywało się z wykorzystaniem identycznych danych wejściowych. W ten sposób uniknięto ewentualnego zafałszowania wyniku poprzez wyeliminowanie czynników

losowych innych niż specyfikacja maszyny testowej.



Wyniki dla algorytmu A* (niedoszacowany):

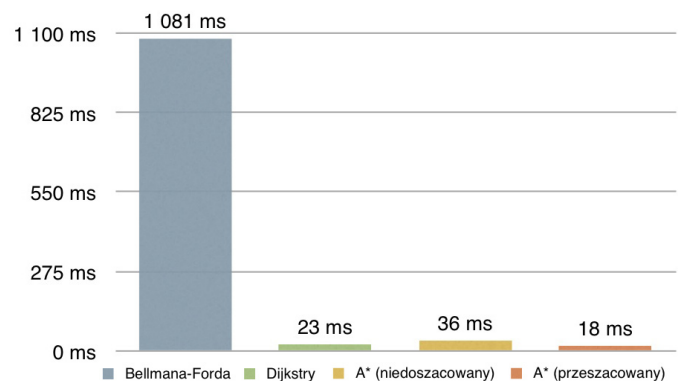
44,45,42,41,40,41,42,40,36,42,39,42,39,39,51,41,39,42,41,43

Rys. 8. Interfejs graficzny aplikacji testowej

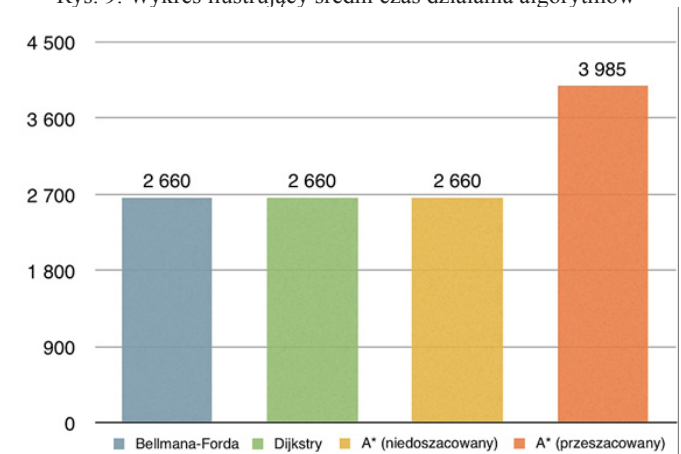
5.2. Wyniki testów

Pierwszym z badanych kryteriów był czas potrzebny do odnalezienia najkrótszej ścieżki przez każdy z programów, mierzony od chwili startu do momentu odnalezienia rozwiązania. Wynik testu został przedstawiony na rysunku 9. Jak widać, najszybszy okazał się algorytm A* (przeszacowany), a następnie algorytm Dijkstry.

Drugim kryterium wyboru był koszt przejścia (suma kosztów wejścia na każde z pól) dla wytyczonych tras. Im niższy koszt znalezionej trasy, tym lepiej. Na rys. 10. przedstawiono wyniki dla każdego z algorytmów. Jak można zauważyć, prawie wszystkie algorytmy odnalazły taką samą trasę.



Rys. 9. Wykres ilustrujący średni czas działania algorytmów



Rys. 10. Wykres ilustrujący koszt znalezionych ścieżek

Z przeprowadzonych badań wynika, że najszybszym algorytmem był algorytm A* (przeszacowany), jednak jednocześnie okazał się najmniej dokładny, jeśli chodzi o koszt znalezionej trasy.

Tabela 2. zawiera podsumowanie wszystkich testów. W kolumnie „Punkty” przedstawiono iloczyn czasu i kosztów. Im niższy wynik, tym lepszy algorytm.

Tab. 2. Podsumowanie wyników testów

| | Średni czas obliczeń | Koszt przejścia | Punkty |
|----------------------------|----------------------|-----------------|-----------|
| Bellmana-Forda | 1081 ms | 2660 | 2 875 460 |
| Dijkstry | 23 ms | 2660 | 61 180 |
| A* (niedoszacowany) | 36 ms | 2660 | 95 760 |
| A* (przeszacowany) | 18 ms | 3985 | 71 730 |

Najlepszym rozwiązaniem dla gier przeglądarkowych wykorzystujących technologię JavaScript okazał się algorytm Dijkstry, który uzyskał wynik lepszy o około 17% od przeszacowanego algorytmu A*, o około 56% od niedoszacowanego i aż o 470% od algorytmu Bellmana-Forda.

6. Wdrożenie najlepszego rozwiązania problemu

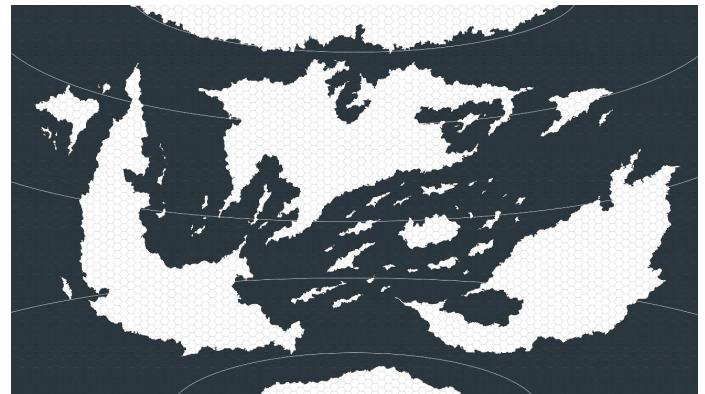
6.1. Projekt systemu

Z rezultatów przeprowadzonych badań wynika, że optymalnym rozwiązaniem jest algorytm Dijkstry. W tym rozdziale zostanie zaprezentowane wykorzystanie tego skryptu w formie rozbudowanej i przystosowanej do współpracy z planszą gry przedstawiającą fikcyjny świat. Plansza została zbudowana z 4900 sześciokątnych pól ułożonych w siatkę o wymiarach 49×100 pól. Pola mogą przedstawiać jeden z trzynastu typów terenu – każdy o innym koszcie ruchu, który symbolizuje trudność poruszania się po danym terenie (im bardziej wymagający, tym wyższy koszt pola).

Dla zróżnicowania kosztów oraz zwiększenia realizmu oprócz typów terenu zastosowano system pogody. Pogoda modyfikuje koszt terenu o pewną symboliczną wartość i tym samym przyczynia się do ułatwienia bądź utrudnienia podróży.

Głównym założeniem przy projektowaniu systemu pogodowego było stworzenie silnika prostego do implementacji, który jednocześnie realistycznie odwzorowywałby występowanie zjawisk atmosferycznych.

W celu osiągnięcia zamierzonego efektu planszę podzielono na sześć stref klimatycznych (rys. 11). Przynależność pola do danej strefy wylicza się na podstawie współrzędnych. W systemie funkcjonuje osiem typów pogody. Każdy z nich posiada modyfikator prawdopodobieństwa wystąpienia w zależności od typu terenu oraz strefy klimatycznej, do której należy pole.



Rys. 11 Podział na strefy klimatyczne

6.2. Aplikacja do zarządzania danymi

Na potrzeby projektu napisano aplikację, która ułatwia zarządzanie danymi. Udostępnia ona trzy tryby pracy: wizualizację danych, edycję terenu oraz generator pogody. Na rys.12. przedstawiono podgląd interfejsu aplikacji.



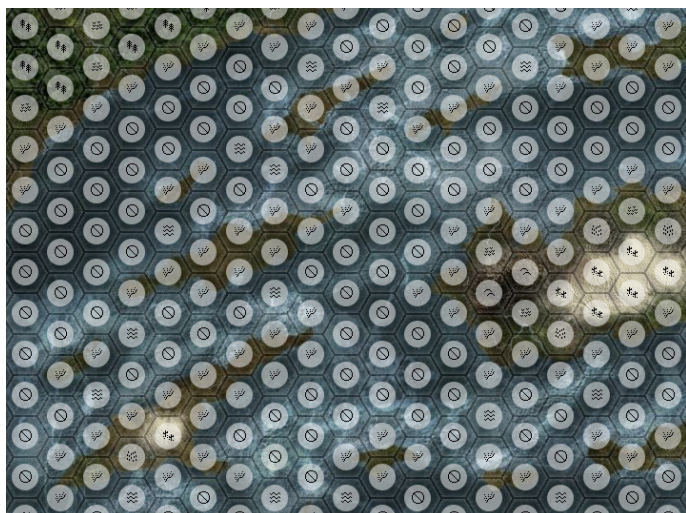
Rys. 12 Ogólny widok interfejsu

Aplikacja pracuje w kilku trybach:

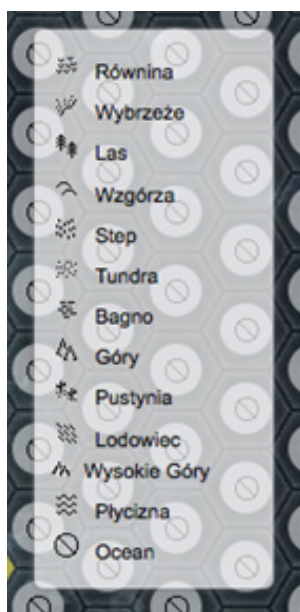
- wizualizacji danych – tryb umożliwi podgląd różnych właściwości pól: typu terenu, pogody panującej na danym polu, kosztu ruchu (składającego się z kosztu terenu i modyfikatora pogodowego) oraz współrzędnych pól.
- edycji terenu – w ramach niego udostępniono użytkownikowi możliwość grupowego zmieniania typu terenu.
- generowania pogody – umożliwi automatyczne generowanie typów pogody zgodnie z założeniami systemu pogodowego

Na rys. 13. przedstawiono fragment widoku z informacjami na temat typu terenu (dla zwiększenia czytelności oznaczenie typów terenów znajduje się na rys. 14.). Użytkownik może przełączać się pomiędzy widokami poprzez wybranie odpowiedniej opcji z przybornika narzędzi (rys. 15.). Szczegółowy opis obsługi poszczególnych trybów aplikacji znajduje się w pracy *Analiza i implementacja [...]*¹¹.

11. T. Sobczak, *op. cit.*



Rys. 13. Przykładowy widok z informacjami na temat typu terenu



Rys. 14. Oznaczenie typów terenu

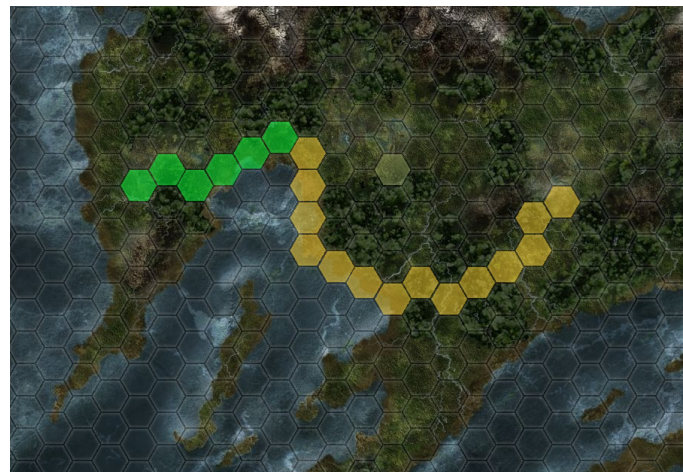


Rys. 15. Przybornik narzędzi dla trybu wizualizacji

6.3. Budowa modułu nawigacji

Skrypt symulujący przemieszczanie się gracza po planszy również został włączony do aplikacji z narzędziami jako dodatkowy tryb. Po jego wybraniu wystarczy wskazać początek i koniec trasy, a następnie wcisnąć przycisk start (↵). Po chwili program narysuje nam wyliczoną trasę, po czym rozpocznie jej animowanie.

Na rys. 16. zaprezentowano przykładową trasę w trakcie animacji. Kolorem żółtym zaznaczono wybraną trasę, zielonym zaś pola, które zostały już oznaczone przez system jako przebyte. Czas przejścia pomiędzy polami wyliczany jest na podstawie kosztu pola docelowego pomnożonego przez jedną dziesiątą sekundy.



Rys. 16. Przykładowa trasa w trakcie animacji

7. Podsumowanie

W artykule przedstawiono trzy najbardziej popularne algorytmy odnajdywania ścieżek. Przeprowadzono analizę porównawczą, która polegała na zestawieniu ze sobą średnich czasów działania oraz zwróconych kosztów tras znalezionych przez wszystkie algorytmy. Na tej podstawie wybrano algorytm Dijkstry, który na co dzień jest bardzo często wykorzystywany przy wyznaczaniu tras dla pakietów danych w sieciach komputerowych (np. w protokole OSPF).

Autorskim projektem jest moduł gry przeglądarkowej odpowiedzialny za nawigację, który wykorzystuje implementację algorytmu Dijkstry do odnajdywania trasy na mapie. Jego główną zaletą jest zróżnicowanie kosztów przejścia pomiędzy polami wynikające z zaimplementowanego systemu typów terenu i pogody. Najważniejszą wadą tego modułu może być sposób uruchomienia procedury nawigacyjnej, który obecnie jest mało realistyczny (trzeba samodzielnie wyznaczyć punkt startowy – w standardowej rozgrywce gracz powinien startować z miejsca, w którym się aktualnie znajduje). Oprócz modułu nawigacyjnego napisano narzędzia ułatwiające zarządzanie modyfikatorami pól planszy.

Projekt w przyszłości można rozwinąć poprzez zwiększenie realizmu podróży, dodając modyfikatory przejść pomiędzy polami – z niektórych terenów łatwiej jest wyjść niż do nich wejść (np. góry, morze). Dodatkowo można przeprowadzić szerszą analizę algorytmu A* w celu odnalezienia optymalnej funkcji heurystycznej, która sprawi, że czas działania algorytmu i zwrócony koszt odnalezionej trasy będzie lepszy niż obecne wyniki algorytmu Dijkstry.

Bibliografia

1. Ross K. A., Wright C. R. B., *Matematyka dyskretna*, Wydawnictwo Naukowe PWN, Warszawa 1996.
2. Sysło M. M., Deo N., Kowalik J. S., *Algorytmy optymalizacji dyskretniej*, Wydawnictwo Naukowe PWN, Warszawa 1995.
3. T. Sobczak, *Analiza i implementacja algorytmów odnajdywania ścieżki do zastosowania w grach przeglądarkowych*, inżynierska praca dyplomowa, Wrocławska Wyższa Szkoła Informatyki Stosowanej, Wrocław 2012.
4. *Algorytmy i struktury danych*, wykład 7:
http://hector.tu.kielce.pl/przedmioty/aisd-wyk/AiSD_w07.pdf, dostępny w internecie [dostęp 30.11.2021]



Zezwala się na korzystanie z *Analiza i implementacja algorytmów odnajdywania ścieżki do zastosowania w grach przeglądarkowych* na warunkach licencji Creative Commons Uznanie autorstwa 4.0 (znanej również jako CC-BY), dostępnej pod adresem <https://creativecommons.org/licenses/by/4.0/pl/> lub innej wersji językowej tej licencji lub którejkolwiek późniejszej wersji tej licencji, opublikowanej przez organizację Creative Commons.

Testowanie użyteczności oprogramowania do nauki języków klasycznych. Case study aplikacji mobilnej *Speak Latin: Learn Latin Language Offline*

*Usability testing of software for learning classical languages. Case study of the mobile application *Speak Latin: Learn Latin Language Offline**

Jakub Zbądzki¹, Anna Rogowska-Wandowicz²

STRESZCZENIE: Celem artykułu jest przedstawienie znaczenia, jakie mają kategoria użyteczności i jej testowanie dla oprogramowania służącego do nauki języków klasycznych, a także wskazanie możliwych defektów, które mogą się pojawiać, i ich źródeł. Jako przykład posłużyła aplikacja mobilna *Speak Latin: Learn Latin Language Offline*. Testy użyteczności przeprowadzono na aplikacji w wersji 1.0.18, w brytyjskiej wersji językowej, na urządzeniu Huawei Mate 10 Lite z systemem Android w wersji 8.1.0. Oprogramowanie testowano pod kątem efektywności i satysfakcji z użytkowania. Zaobserwowano pojawianie się istotnych defektów w kluczowych obszarach aplikacji. Usterki dotyczą przede wszystkim: 1. nieprawidłowej wymowy wyrazów; 2. braku deklarowanego rozróżniania zapisu i wymowy; 3. schematu odtwarzania nagrań istotnie utrudniającego naukę; 4. niepoprawnych i niejednorodnych tłumaczeń lub ich braku; 5. nieczytelnych kategorii, w których występują dane słowa. Zauważono, że istotna część defektów wynika z zastosowania syntezy i tłumacza mimo słabego rozwinięcia tych technologii w kontekście języków klasycznych. Aplikacja prawdopodobnie nie przeszła też testów użyteczności. Kontrola ze strony człowieka i weryfikacja jakości tego rodzaju oprogramowania okazują się niezbędne. Bez nich nie może ono spełniać wymagań przeciętnych odbiorców, które dotyczą efektywności i satysfakcji z użytkowania, nie jest też w stanie przynieść istotnych korzyści edukacyjnych.

ABSTRACT: The aim of this article is to present the importance of the usability category and its testing for software designed for learning classical languages, as well as to point out possible defects that may appear and their sources. The mobile application *Speak Latin: Learn Latin Language Offline* was used as an example. Usability testing was carried out on the app version 1.0.18, in the British English language version, on a Huawei Mate 10 Lite device with the Android system in version 8.1.0. The software was tested in terms of effectiveness and satisfaction of use. Significant defects have been observed in key areas of the application. The defects relate primarily to: 1. incorrect pronunciation of words; 2. lack of claimed differentiation between notation and pronunciation; 3. the playback pattern of recordings significantly impeding learning; 4. incorrect and inconsistent translations or lack thereof; 5. illegible categories in which words occur. It was noted that a significant proportion of the defects are due to the use of synthesizer and translator despite the poor development of these technologies in the context of classical languages. The application probably did not pass usability testing either. Human control and quality verification of this kind of software prove essential. Without them, it cannot meet the average user's requirements regarding the effectiveness and satisfaction of use, nor is it able to deliver significant educational benefits.

SŁOWA KLUCZOWE: nauka języka obcego, język łaciński, testy użyteczności, user experience, zapewnianie jakości

KEYWORDS: language learning, Latin language, usability testing, user experience, quality assurance

1. Wprowadzenie

Jednym z czynników decydujących o powodzeniu podczas nauki języków obcych jest skuteczność wykorzystywanych narzędzi edukacyjnych. Za sprawą powszechnej cyfryzacji coraz większą rolę w tym procesie będzie odgrywać oprogramowanie. Obecnie ze względu na wysoką dostępność

urządzeń mobilnych znaczną popularność zyskują przeznaczone na nie aplikacje do celów (auto)dydaktycznych – począwszy od kompleksowych takich jak HelloTalk, przez pomocnicze w rodzaju Quizlet, aż po aplikacje niesłużące bezpośrednio do nauki, ale wykorzystywane w jej trakcie,

1. Uniwersytet Wrocławski, 50-137 Wrocław, e-mail: jakub.zbadzki@uwr.edu.pl, ORCID 0000-0002-6268-5469.

2. Uniwersytet Wrocławski, 50-137 Wrocław, e-mail: anna.rogowska-wandowicz@uwr.edu.pl, ORCID 0000-0002-7331-9608.

jak słowniki czy translatory³. Najnowsze badania wskazują, że zaawansowane programy na urządzenia stacjonarne i mobilne pozwalają nie gorzej niż tradycyjne metody realizować złożone cele: umożliwiają choćby w angażujący sposób usprawniać przyswajanie drugiego języka⁴, są też wykorzystywane z powodzeniem do specjalistycznych zastosowań, takich jak kształcenie sprawności pisania w kontekście akademickim⁵.

2. Nowoczesna nauka łaciny

Równoległe do opisywanej rewolucji technologicznej toczy się proces zmian w dydaktyce języków klasycznych, szczególnie w obszarze łaciny. Trzeba tu wyjaśnić, że specyficzna tradycja nauczania tego języka przez długie lata nie dopuszczała osiągnięć nowoczesnej glottodydaktyki języków żywych do sali szkolnej (czy uniwersyteckiej). Dopiero od niedawna wśród lektorów języka łacińskiego wzrasta zainteresowanie sposobami nauczania innymi niż metoda gramatyczno-tłumaczeniowa, koncentrująca się na wykształceniu w adepcie języka przede wszystkim umiejętności czytania tekstów. Droga do zrozumienia słowa pisanego prowadzi przy tym poprzez meandry skomplikowanych zagadnień gramatycznych, które należy opanować pamięciowo przed właściwym kontaktem z autentycznym tekstem, a także przez tłumaczenie na język ojczysty. Przekład, przez współczesnych glottodydaktyków zasadniczo uważany za element spowalniający akwizycję języka⁶, w opisywanej metodzie pełni więc rolę kluczową. Taki sposób nauczania jeszcze w XIX w. był praktykowany również na polu języków żywych, jednak wobec jego ograniczonej skuteczności stopniowo zaczęto wypracowywać inne podejścia. Obecnie dąży się do wykształcenia w osobach uczących się języka dwóch sprawności receptywnych (czytania i słuchania) oraz dwóch produktywnych (mówienia i pisania). W nauczaniu języków żywych bardzo często stosuje się podejście komunikacyjne (*communicative approach*), w którym kładzie się nacisk na komunikację uczniów w języku docelowym. W tym miejscu warto dodać, że nauczanie łaciny przez wieki odbywało się w warunkach immersji językowej – taka sytuacja miała miejsce przed przełomem oświeceniowym, kiedy miała ona silną pozycję języka międzynarodowego oraz głównego języka wykładowego w szkołach⁷. Później jednak zaczęto traktować mowę Rzymian, obok

pozostałych komponentów dziedzictwa starożytności, jako narzędzie kształtowania myślenia i charakteru. Od tej pory miejsce konwersacji w języku docelowym na dobre zajęło wymaganie od wychowanków wyuczenia się tabelek gramatycznych i słówek na pamięć. Łaciny przestał też używać sam nauczyciel. Obrazy uczniów udręczonych tak pojętym treningiem umysłowym można odnaleźć choćby w polskich powieściach doby pozytywizmu. Z czasem niektórzy zaczęli zauważać wady opisywanego podejścia. Jedną z pierwszych propozycji zmian była utworzona w Wielkiej Brytanii przez Williama Henry'ego Denhama Rouse'a metoda bezpośrednia (*direct method*). Prawdopodobnie przy opisie właśnie tego zjawiska pierwszy raz użyto w kontekście dydaktyki wyrażenia „łacina żywa”⁸. Określenie to bywa różnie interpretowane, co prowadzi do licznych nieporozumień. Jak wskazuje Marcin Loch, tą nazwą bywają określane metody dydaktyczne przeciwstawione metodzie gramatyczno-tłumaczeniowej, ale też techniki urozmaicania „tradycyjnych” zajęć elementami interakcyjnymi, a przede wszystkim podejmowana współcześnie komunikacja w języku łacińskim poza salą lekcyjną, na przykład tworzenie tekstów o różnym charakterze⁹. Obecnie łacina żywa pojmowana na wszystkie wymienione sposoby przeżywa rozkwit. Niewątpliwie sprzyja temu kierunek rozwoju technologicznego. Narzędzia takie jak internet, umożliwiające niezwykle szybki przepływ informacji, pozwalają na szersze rozpowszechnianie treści, które wcześniej miały zasięg lokalny. Nowo powstające ośrodki, które proponują nauczanie zgodne z podejściem komunikacyjnym, zyskują rozgłos i co roku przyciągają wielu chętnych na letnie kursy językowe¹⁰. W czasach pandemii COVID-19 wiele z nich odbywa się zdalnie. W sieci ma miejsce także wymiana doświadczeń pomiędzy nauczycielami i osobami uczącymi się języka, również samodzielnie. Internet umożliwił też swobodne prowadzenie rozmów w czasie rzeczywistym przez adeptów łaciny z różnych zakątków świata (ze względu na wąskie zainteresowanie językami klasycznymi nierzadko trudno o spotkanie dwóch osób chętnych na łacińskie konwersacje w jednym miejscu). Oryginalne treści tworzone po łacinie coraz częściej pojawiają się także na platformach takich jak YouTube czy Facebook. Dostęp do metod akwizycji języka alternatywnych dla najczęściej proponowanej przez szkoły i uczelnie wyższe metody gramatyczno-tłumaczeniowej z pewnością będzie

3. Wg przeglądu i podziału zaproponowanego w publikacji F. Rosell-Aguilar, *State of the App: A Taxonomy and Framework for Evaluating Language Learning Mobile Applications*, „Calico” 2017, nr 2, s. 248–251.

4. E. Rusman, S. Ternier, M. Specht, *Early Second Language Learning and Adult Involvement in a Real-World Context: Design and Evaluation of the „ELENA Goes Shopping” Mobile Game*, „Educational Technology & Society” 2018, nr 3, s. 90–103.

5. C. C. Lin, G. Z. Liu, T. I. Wang, *Development and Usability Test of an e-Learning Tool for Engineering Graduates to Develop Academic Writing in English*, „Educational Technology & Society” 2017, nr 4, s. 148–161.

6. Hanna Komorowska pośród niepożądanych skutków nadużywania przez nauczycieli tłumaczenia wymienia między innymi „traktowanie języka obcego przez ucznia tak, jakby był dosłownym tłumaczeniem z języka polskiego” oraz „zwolnione tempo nauki, które jest naturalną konsekwencją tego, iż znaczne fragmenty lekcji toczą się w języku polskim”. H. Komorowska, *Metodyka nauczania języków obcych*, Fraszka Edukacyjna, Warszawa 2009, s. 154.

7. Historię nauczania łaciny – z uwzględnieniem obecności komunikacji w języku docelowym w procesie nauczania – zwięźle przedstawia Katarzyna Ochman, zob. K. Ochman, *Nauczanie łaciny. Immersja czy gramatyka?*, „Języki obce w szkole” 2014, nr 3, s. 54–58.

8. M. Loch, *Żywa łacina — między prawdą a mitem*. Zarys historii zagadnienia, „Language and Literary Studies of Warsaw” 2013, nr 3, 153–170.

9. *Ibidem*.

10. Na przykład organizowana w Poznaniu Schola Aestiva Posnaniensis. Podobne przedsięwzięcia podejmują między innymi włoska Accademia Vivarium Novum, amerykańska Paideia czy University of Kentucky.

skutkowało coraz większym zapotrzebowaniem na nowoczesne narzędzia umożliwiające łatwiejsze przyswojenie łaciny. Ich skuteczność nie zależy jednak wyłącznie od zastosowania najnowszych technologii.

3. Użyteczność

Warunkiem efektywności oprogramowania jest spełnienie wymogów dotyczących użyteczności (ang. *usability*). Sam termin jest często utożsamiany z tzw. *user experience* (UX), czyli z ogółem wrażeń, jakie wywiera na użytkownika dany program. Bez wątpienia użyteczność zawiera się w UX, oznacza jednak rzecz węższą. Określająca je norma ISO 9241 wyodrębnia trzy aspekty istotne dla osiągnięcia przez użytkownika zamierzonego celu: wydajność, efektywność i satysfakcję czerpaną z kontaktu z produktem¹¹ – i tak będziemy ją rozumieć na potrzeby tej pracy. Problem użyteczności dotyczy szczególnie aplikacji mobilnych, co wynika zapewne z nagminnego mechanicznego przenoszenia na nie rozwiązań z urządzeń stacjonarnych. Braki w tym zakresie są uznawane za najważniejszy czynnik, z powodu którego użytkownicy rezygnują z korzystania z danego programu¹². Tego rodzaju niedostatki są też uznawane za jeden z głównych problemów odpowiedzialnych za niedostateczne wykorzystywanie potencjału narzędzi cyfrowych podczas nauki¹³. Użyteczność oprogramowania, stając się coraz częściej istotną pomocą naukową czy wręcz jedynym narzędziem edukacyjnym, zaczyna się więc stawać jednym z aspektów, które wymagają rzetelnej walidacji ze strony specjalistów zajmujących się QA. Defekty w tym obszarze mogą wpływać na budowanie złych nawyków, wprowadzać użytkownika w błąd, a nawet niweczyć sukces edukacyjny. Jest to szczególnie istotne w przypadku języków klasycznych, ponieważ efekty rozwoju technologii nie osiągnęły ich w równym stopniu co języków nowożytnych. Ponieważ oprogramowanie do ich nauki będzie się pojawiać coraz częściej, warto przyjrzeć się potencjalnym defektom, jakie mogą w nim występować, oraz wskazać, z czego one wynikają i jak zapobiegać im w przyszłości.

3. Speak Latin – analiza użyteczności

W badaniach wykorzystamy *Speak Latin: Learn Latin Language Offline* w wersji 1.0.18. Posłużymy się jej brytyjską wersją językową ze względu na jej najszersze rozpowszechnienie. Aplikacja zostanie przetestowana na urządzeniu Huawei Mate 10 Lite z systemem Android w wersji 8.1.0. Program, jak wskazuje jego nazwa, zdaje się obiecywać bardzo wiele. Wbrew oczekiwaniom nie jest on jednak osadzony całkowicie w nurcie łaciny żywej.

To raczej aplikacja dość ogólnego przeznaczenia skierowana do osób chcących poznać poprawną wymowę i poszerzyć swój zasób słownictwa, a także pogłębić znajomość łaciny współczesnej. Wyrazy są pogrupowane wedle kategorii tematycznych i gramatycznych. Zapamiętywaniu mają sprzyjać dodane do nich grafiki oraz nagrania audio. Program oferuje też kilka edukacyjnych gier takich jak dopasowanie wyrazu do ilustracji lub kategorii.

Pierwszemu uruchomieniu aplikacji może towarzyszyć pewne zdziwienie. Wydawałoby się, że ze względu na swój cel będzie ona perfekcyjnie dopracowana pod względem zapisu. Menu służące do wyboru języka zawiera jednak usterki w postaci błędnie zapisanych nazw ze znakami diakrytycznymi: po literach, które je zawierają, kolejne z nich nieoczekiwanie zwiększają rozmiar. To najpewniej przejaw problemu z kodowaniem. Mimo że jest natury technicznej, a nie filologicznej, nie powinien wystąpić w aplikacji służącej do nauki języka, w której przypadku dokładny zapis jest niezbędny. O ile w większości sytuacji tego rodzaju usterki nie będą przesądzać o efektywności produktu (zapisy nie są mylące, a jedynie nieestetyczne), o tyle bez wątpienia wpływają negatywnie na satysfakcję związaną z jego użytkowaniem.



Rys. 1. Przykład problemów ze znakami diakrytycznymi

Jak wskazano, aplikacja jest skierowana do osób chcących poznać prawidłowe brzmienie wyrazów. Wydaje się, że w zamierzeniu twórców program miał oddawać kościelną (włoską) wymowę łaciny, co słychać w takich słowach jak *niger* ('ni.dʒer) czy *cinereus* (tʃi'ne.re.us)¹⁴. Niestety nie zastosowano żadnego określonego sposobu wymowy (restytuowanej lub narodowej); co więcej, nie wykorzystano nawet pomocy lektora. Prawdopodobnie sięgnięto po syntezytor naśladowujący sam język włoski, co da się wywnioskować na podstawie błędnie odczytywanych słów takich jak *caeruleus*. W łacinie kościelnej brzmiałoby ono

11. ISO 9241-210:2010(E). Ergonomics of Human-System Interaction – Part 210: Human-Centred Design for Interactive Systems.

12. H. Hoehle, V. Venkatesh, *Mobile Application Usability: Conceptualization and Instrument Development*, „MIS Quarterly” 2015, nr 2, s. 436.

13. G. Mesfin, G. Ghinea, T. Grønli, W. H. Hwang, *Enhanced Agility of E-Learning Adoption in High Schools*. „Educational Technology & Society” 2018, nr 4, s.160.

(tʃe'ru.le.us, zgodnie z dwiema zasadami: uproszczeniem dyftongu *ae* do *e* oraz wymawianiem *c* przed *e* jako (tʃ, nie zaś *k*. Syntezator odczytuje jednak wyraz jako ka.e.ru.le.us., co odpowiada logice kierującej językiem włoskim (*c* przed *a* jako *k*, brak dyftongów). Ponadto aplikacja nie spełnia zamierzonego zadania w postaci podawania zarówno zapisu, jak i brzmienia wyrazów. Podawanie identycznych form mimo deklaracji o ich rozróżnianiu wprowadza użytkowników w błąd, ponieważ sugeruje ich zbieżność, podczas gdy w wielu przypadkach tak nie jest. Jako przykład może posłużyć wspomniany wcześniej *caeruleus* (w wymowie restytuowanej kae'ru.le.us, w kościelnej tʃe'ru.le.us).

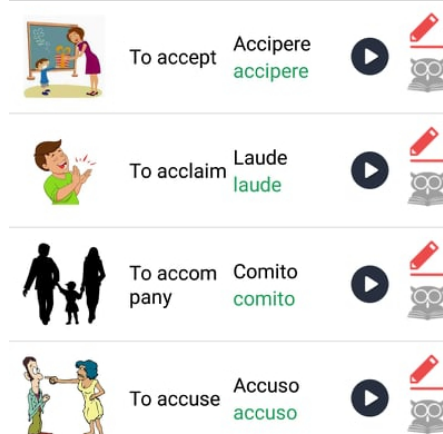


Rys. 2. Przykład braku rozróżnienia zapisu i wymowy

Zastosowanie syntezatora znacząco odbiega od norm przyjętych w podobnych programach do nauki języka korzystających z lektorów, zwłaszcza że stosuje się on do innego języka niż ten, którego ma uczyć aplikacja. Jest to wbrew podstawowemu celowi aplikacji, który stanowi poznanie i utrwalenie słów w prawidłowym brzmieniu. W tym wypadku nie może on zostać spełniony, radykalnie obniżając efektywność oprogramowania. Oczywiście w przypadku łaciny nie można sięgnąć po pomoc *native speakerów*, jednak osób znających ją na poziomie wystarczającym do prawidłowego odczytania słów jest na tyle dużo, że konsultacja z nimi nie stanowiłaby żadnego kłopotu.

Wiele defektów można zaobserwować także w pomocniczym dla nauki wymowy obszarze dotyczącym nagrań. Poszczególne wyrazy w aplikacji są zamieszczone w osobnych kategoriach, np. *Colours* czy *Adjectives*. Podczas odsłuchiwania listy słów z danej sekcji po kliknięciu *play*, *stop* i ponownie *play* aplikacja odtwarza wszystkie wyrazy z niej od początku. Typowy użytkownik oczekiwałby tymczasem rozpoczęcia od ostatniego zaznaczonego elementu, który chciałby utrwalić. Dla osoby zmuszonej do czekania, aż aplikacja dojdzie z powrotem od początku do wskazanego miejsca, usterka ta skutkuje stratą czasu wywołującą zniecierpliwienie. Uruchomienie od nowa wszystkich nagrań z kategorii jest bezcelowe, jeśli zawiera ona w większości wyrazy już opanowane. Identyczna usterka dotyczy przycisku *repeat times* działającego tylko na całą playlistę. Pojawia się też defekt zwią-

zany z czasem opóźnienia pomiędzy kliknięciem przycisku odtwarzania nagrania a jego uruchomieniem. Jest on niejednakowy dla różnych słów. Co więcej, często jest też krótszy niż czas potrzebny na wypowiedzenie słowa na głos lub w myślach. Zastosowanie większej przerwy dałoby szansę na działanie ze strony użytkownika, wpływając pozytywnie na proces zapamiętywania. Problem dotyczy wyrazów krótszych niż dwusylabowe, trudno jednak określić, dlaczego to akurat one są odtwarzane z mniejszym opóźnieniem. Defekty dotyczące nagrań nie niwelują całkowicie użyteczności aplikacji, ale mogą obniżać jej efektywność, a już na pewno przekładają się na zmniejszoną satysfakcję użytkownika. Warto zwrócić na to uwagę, zwłaszcza że w testach pomiary czasu zwykle wykonuje się w kontekście wydajności systemu, a rzadziej bada się subiektywne doświadczenia związane z czasem interakcji, choć również one są bardzo istotne¹⁵. Następną kategorią defektów o szerokim zasięgu obejmującą zagadnienia związane z tłumaczeniem wyrazów. Istotną część z nich została przełożona nieprawidłowo. Rodzaje usterek są różne. Pojawia się m.in. pomieszczenie części mowy – rzeczownik *smile* został oddany jako imiesłów przymiotnikowy czynny w liczbie mnogiej *subridentes* (śmiejący się). Formy czasowników nie są ujednolicone. Po angielsku, naturalnie, podano je zawsze w bezokoliczniku (*to accept*, *to accuse*, *to attack*), ale w przypadku łaciny nie przyjęto żadnej jednolitej zasady, która by je uporządkowała (wobec czego podano kolejno *accipere*, *accuso*, *impetum* – a więc bezokolicznik czasu teraźniejszego, strony czynnej; 1. os. l. poj., tryb oznajmujący czasu teraźniejszego, strony czynnej; rzeczownik w accusativie i l. poj. – zamiast czasownika *impetere*).



Rys. 3. Przykład braku ujednoliconych form czasowników

Niekiedy błąd w tłumaczeniu wynika z istnienia homonimu w języku, z którego dokonuje się przekładu. Angielskie słowo *bank* oznacza zarówno instytucję, jak i brzeg

14. Wraz z rozszerzaniem się rzymskiego imperium powiększał się obszar, na którym używano języka łacińskiego. Na przestrzeni lat w różnych miejscach łacina podlegała rozmaitym procesom fonetycznym, co doprowadziło do powstania tak zwanych wymów narodowych, zwykle posiadających cechy wymowy łaciny z czasów późnej starożytności (zanik iloczasu, monofongizacja) i pozostających pod wpływem języków używanych na danym terenie. Czytanie tekstów starożytnych wymową niezgodną z tą, którą posługiwali się autorzy antyczni, budzi pewne wątpliwości, szczególnie w przypadku utworów poetyckich, dlatego podejmuje się wysiłki odtworzenia dawnej fonetyki języka łacińskiego, m.in. na podstawie zapisów inskrypcyjnych. Obecnie coraz więcej osób, posługuje się wymową restytuowaną opartą na rekonstrukcji wymowy łacińskiej przełomu er. W kontekście kościelnym standardem pozostaje wymowa włoska.

15. A. K. Trapp, N. Backhaus, R. Thomaschke, *Measuring subjectively experienced time in usability and user experience testing scenarios*, „International Journal of Human-Computer Studies” 2020, nr 138, s. 3.

czy wybrzeże. Zostało ono na łacinę przełożone jako *ripa* (a więc w drugim znaczeniu), mimo że kontekst – obrazek z bankiem – wskazuje, że chodziło o to pierwsze. Podobnie wyraz *back*, po angielsku plecy lub kierunek – do tyłu, przetłumaczono jako *retro* (ponownie w drugim sensie), podczas gdy grafika przedstawia część ciała. Niektóre z wyrazów pojawiają się w nieistniejącej czy nieprawidłowej formie gramatycznej.

Pierwszy rodzaj usterki dotyczy np. *Sunday* (*solis dies* zamiast *solis dies*). Początkujący uczeń może wywnioskować, że ma do czynienia ze słowem należącym do drugiej deklinacji w rodzaju męskim zakończonym w genetywie na -i, podczas gdy bez wątpienia odmienia się ono zgodnie z regułami deklinacji III i ma końcówkę -is. Forma *solis* na oznaczenie Słońca nie funkcjonuje w łacinie. Drugi rodzaj defektu obejmuje wyrazy podawane w niewłaściwym przypadku czy jako inna część mowy. Dla przykładu: słowo *economics* w kategorii *Education & Learning* oddano jako *oeconomicam* (dosł. „ekonomiczną”), a nie *oeconomia*, a zatem jako przymiotnik w accusative zamiast jako rzeczownik w nominative.

Nie stanowi to dużego kłopotu dla osób co najmniej średnio zaawansowanych, ponieważ i tak łatwo im przyjdzie odgadnąć właściwą formę, ale dla początkujących może to być co najmniej mylące, a z pewnością utrudnia naukę. Część słów została też przetłumaczona niedokładnie – tak jest z *cash* oddanym jako *nummum* (a więc nie „pieniądze”, a „monetę” – metonimicznie, podano też nie nominativus, ale accusativus). Na marginesie tej kategorii można także wskazać problem w postaci braku przełożenia niektórych wyrazów – np. *bookstore* w kategorii *Town*.



Rys. 4. Przykład brakującego tłumaczenia

Skąd wynikają tego rodzaju defekty? Aplikacja korzysta z translatora, najprawdopodobniej Google Translate – dowodzi tego własnoręczna próba przetłumaczenia z jego użyciem wyrazów zamieszczonych w aplikacji (co prawda nie na całym korpusie wyrazów, więc jest to wnioskowanie indukcyjne). Wskazują na to także brakujące słowa, które nie funkcjonowały pierwotnie w łacinie i próżno o nie w translatorach, natomiast da się je utworzyć (w przypadku *bookstore* mogłaby to być np. *taberna libraria* – za *Małym słownikiem polsko-łacińskim* Lidii Winniczuk¹⁶). Oprogramowanie do przekładu dzięki uczeniu maszynowemu przeszły nadzwyczajną przemianę, przez co oferuje obecnie często zupełnie przyzwoite przekłady, wystarczające do celów amatorskich. Nie dotyczy to jednak języków starożytnych, w przypadku których ma ono wielokrotnie mniejsze zasoby potrzebne do procesu maszynowego uczenia się, przez co jakość ich cyfrowego tłumaczenia

jest nader niska. Niestety żaden program służący do nauki języka nie może mieć takich defektów, jeśli ma być użyteczny, rzutują one bowiem bezpośrednio na jego efektywność, a wręcz przekreślają efekty korzystania z niego. W *Speak Latin* pojawiają się również pewne poboczne usterki dotyczące przyporządkowania słów do właściwych kategorii. Aby nauczyć się danego wyrazu, trzeba wybrać w aplikacji odpowiednią z nich. Niestety niektóre z nich nachodzą na siebie znaczeniowo. Problem dotyczy przede wszystkim jedzenia i sekcji *food fruit*, *food misc* (jedzenie różne), *food non-veg*, *food veg* i *seafood*. Czasami taki podział może sprawiać trudność. Czy makaron i ryż to jedzenie różne czy wegańskie? Oczywiście zależy to od kontekstu, jednak pierwsze z nich zostało podpięte pod kategorię *food misc*, a drugie *food veg*. Do tej ostatniej należy też np. cytryna, choć bardziej pasowałaby do owoców. Podobnych, nieintuicyjnych rozwiązań jest dość dużo. Ponadto pojedyncze kategorie są niejasno opisane. Dotyczy to głównie sekcji *Time* i *Time II*. Gdzie znajdzie się tłumaczenie słowa *minute*? Użytkownik nie może tego wiedzieć przed ich wybraniem. Musi także zapamiętać, w której z nich znajdowało się dane słowo, jeśli chce do niego wrócić. Oba te problemy nie obniżają znacząco użyteczności oprogramowania, ale z pewnością wpływają negatywnie na satysfakcję z jego użytkowania.

4. Podsumowanie

W obecnym kształcie z uwagi na dużą liczbę defektów użyteczności *Speak Latin* wydaje się programem ze zmarnowanym potencjałem. Przyjęcie dla łaciny wymowy jak dla języka włoskiego (w dodatku niesłusznie prezentowanej jako identyczna z zapisem) oraz różnego rodzaju błędne tłumaczenia czy wręcz ich brak przekładają się na bardzo ograniczoną użyteczność aplikacji. Obecność usterek tak znaczącej wagi wynika w dużej mierze z zastosowania syntezatora i translatora oraz wynikających stąd konsekwencji. Być może nie byłoby to najgorsze (choć wciąż złe) rozwiązanie w przypadku oprogramowania służącego do nauki języków nowożytnych, ale w przypadku tych klasycznych stanowi znaczącą przeszkodę. Jest to istotna lekcja na przyszłość: osób szukających podobnych programów będzie przybywać. Nie można jednak w procesie wytwarzania oprogramowania polegać na mechanicznym zastosowaniu narzędzi wykonujących bardzo złożone procesy związane z wykonywaniem operacji na języku mowy. Tam, gdzie niezbędna jest stuprocentowa poprawność, wciąż okazuje się potrzebna pomoc ze strony ludzi – oraz wykonywanie testów użyteczności, które zniwelują ewentualne defekty.

16. *Mały słownik polsko-łaciński*, red. L. Winniczuk, Wydawnictwo Naukowe PWN, Warszawa 1994, s. 232.

Bibliografia

1. H. Hoehle, V. Venkatesh, *Mobile Application Usability: Conceptualization and Instrument Development*, „MIS Quarterly” 2015, nr 2, s. 435–472.
2. ISO 9241-210:2010(E). Ergonomics of Human-System Interaction - Part 210: Human-Centred Design for Interactive Systems.
3. Komorowska H., *Metodyka nauczania języków obcych*, Fraszka Edukacyjna, Warszawa 2009.
4. Lin C. C., Liu G. Z., Wang T. I., *Development and Usability Test of an e-Learning Tool for Engineering Graduates to Develop Academic Writing in English*, „Educational Technology & Society” 2017, nr 4, s. 148–161.
5. Loch M., *Żywa łacina – między prawdą a mitem. Zarys historii zagadnienia*, „Language and Literary Studies of Warsaw” 2013, nr 3, s. 153–170.
6. *Mały słownik polsko-łaciński*, red. L. Winniczuk, Wydawnictwo Naukowe PWN, Warszawa 1994.
7. Mesfin G., Ghinea G., Grønli T., W. H. Hwang, *Enhanced Agility of E-Learning Adoption in High Schools*. „Educational Technology & Society” 2018, nr 4, s. 157–170.
8. Ochman K., *Nauczanie łaciny. Immersja czy gramatyka?*, „Języki obce w szkole” 2014, nr 3, s. 54–58.
9. Rosell-Aguilar F., *State of the App: A Taxonomy and Framework for Evaluating Language Learning Mobile Applications*, „Calico” 2017, nr 2, s. 234–258.
10. Rusman E., Ternier S., Specht M., *Early Second Language Learning and Adult Involvement in a Real-World Context: Design and Evaluation of the „ELENA Goes Shopping” Mobile Game*, „Educational Technology & Society” 2018, nr 3, s. 90–103.
11. Trapp A. K., Backhaus N., Thomaschke R., *Measuring subjectively experienced time in usability and user experience testing scenarios*, „International Journal of Human-Computer Studies” 2020, nr 138, s. 1–46.

pl/ lub innej wersji językowej tej licencji lub którejkolwiek późniejszej wersji tej licencji, opublikowanej przez organizację Creative Commons.



Zezwala się na korzystanie z *Testowanie użyteczności oprogramowania do nauki języków klasycznych. Case study aplikacji mobilnej Speak Latin: Learn Latin Language Offlinena* warunkach licencji Creative Commons Uznanie autorstwa 4.0 (znanej również jako CC-BY), dostępnej pod adresem <https://creativecommons.org/licenses/by/4.0/>

Wprowadzenie do modelowania w języku UML

Introduction to UML modeling

Konrad Szynalski¹, Dawid Różański²

STRESZCZENIE: UML, czyli Zunifikowany Język Modelowania, służy do zapisywania projektu systemu i może być również stosowany w celu graficznego opracowania lub tworzenia oprogramowania. Umożliwia on konstruowanie diagramów, które przedstawiają różne punkty widzenia systemu. W obecnych czasach UML stosuje się również w innych branżach, ponieważ przy jego pomocy można w komfortowy sposób analizować oraz modelować różne działania. Jedną z największych zalet tego języka jest fakt, że pozwala na jednakową interpretację wszystkich modeli przez osoby, które się nim zajmują. Co ważne, może być on zapisywany w formie graficznej przystępnej dla większości osób oraz w formie kodu ukierunkowanego na programistów.

ABSTRACT: UML or Unified Modeling language is used to save the system design and can also be used for graphic development or software development. It allows to create diagrams that present different points of view of the system. Nowadays, UML is also used in other industries because it allows to conveniently analyze and model various activities. One of the greatest advantages of this language is the fact that it makes it possible to equally interpret all the models by the people who deal with it. Importantly, it can be saved in a graphical form that is accessible to most people and in the form of code that is aimed at programmers.

SŁOWA KLUCZOWE: UML, modelowanie, zunifikowany język modelowania

KEYWORDS: UML, modeling, unified modeling language

1. Wprowadzenie

UML, czyli Zunifikowany Język Modelowania, to graficzny język służący do wizualizowania, specyfikowania, konstrukcji i dokumentacji elementów związanych z tworzeniem systemów informatycznych³. Umożliwia standaryzację sposobu opracowywania przekrojów systemu obejmujących różnorodne obiekty takie jak funkcje systemowe czy schematy baz danych.

Sam język łączy w sobie najlepsze cechy z obszarów:

- modelowania danych (diagram związków encji),
- modelowania przepływów (diagram przepływu danych),
- modelowania obiektowego (analiza obiektowa),
- zarządzania złożonością (komponenty).

Początkowo głównym przeznaczeniem Zunifikowanego Języka Modelowania była budowa systemów informatycznych. Okazało się jednak, że UML jest bardzo dobrą formą przekazu informacji, więc zaczęto go również wykorzystywać w innych dziedzinach takich jak transport, telekomunikacja, przemysł obronny i lotniczy czy usługi bankowe. Zazwyczaj UML jest stosowany wraz ze swoją reprezentacją graficzną – jego elementom przypisane są symbole, które

wiąże się na diagramach. Dzięki temu opracowane schematy pozwalają na dokładne śledzenie procesów. W aktualnej wersji UML oznaczonej numerem 2.5.1 można wyróżnić 13 rodzajów diagramów głównych oraz 3 typy interakcji⁴:

- diagramy struktury: diagramy klas, obiektów, komponentów, struktur złożonych, pakietów, wdrożenia;
- diagram zachowania: diagramy przypadków użycia, maszyny stanowej, czynności;
- diagramy interakcji: diagramy przeglądu interakcji, sekwencji, komunikacji, czasowe.

UML nie jest językiem programowania graficznego, ale modele w nim zapisane mogą być powiązane z wieloma językami programowania. Warto zaznaczyć, że model utworzony w języku UML można przekształcić, np. w tabelę relacyjnej bazy danych. Oczywiście działa to również w drugą stronę i na podstawie implementacji da się stworzyć model graficzny. Co ważne, szczegółowość diagramów zależy od fazy tworzenia oprogramowania. Do analizy problemu można stworzyć diagram ogólny, a przy dokumentacji technicznej należy skorzystać z diagramu z większą liczbą szczegółów.

1. Wrocławska Wyższa Szkoła Informatyki Stosowanej, e-mail: konrad.szynalski@gmail.com, ORCID 0000-0001-9367-8088.

2. Wrocławska Wyższa Szkoła Informatyki Stosowanej, e-mail: drozanski@outlook.com, ORCID 0000-0002-9394-120X.

3. *Unified Modeling Language*, <https://it.pwn.pl/Artykuly/Zarzadzanie-projektami/Unified-Modeling-Language-UML>, dostępny w internecie [dostęp 4.12.2021]

4. *UML*, <https://mfiles.pl/pl/index.php/UML>, dostępny w internecie [dostęp 4.12.2021].

2. Standardy języka UML

W każdym języku istnieją pewne standardy i tak też jest w przypadku UML. Każdy obiekt reprezentowany jest przez jeden prostokąt, w którym zawierają się jego składniki. Prostokąt służy do umieszczania w nim nazw elementów, nazw wraz z polem lub nazw z polem i metodą. Do składników elementów można dołączyć również informacje o typie, typie zwracanym lub argumentach. Warto dodać, że składniki klasy mogą posiadać różne modyfikatory dostępu⁵:

- + jest składnikiem publicznym (public)
- # jest składnikiem chronionym (protected)
- jest składnikiem prywatnym (private)
- ~ jest składnikiem dostępnym w obrębie projektu (package)

2.1. Reprezentacja klas abstrakcyjnych i interfejsów UML

Do oznaczenia metod abstrakcyjnych, które znajdują się w klasie abstrakcyjnej, używa się kursywy tudzież podkreślenia. Sam interfejs przedstawia się tak jak klasy, lecz jego nazwa musi zostać poprzedzona słowem kluczowym `<<interface>>`. Co więcej, implementację interfejsu do danej klasy należy oznaczyć pustym białym grotem strzałki, który umieszcza się na końcu przerywanej linii. Należy pamiętać, że klasa implementująca interfejs musi wdrożyć jego metody. Jeśli chodzi o klasy abstrakcyjne, w tym przypadku klasa dziedzicząca musi implementować metody abstrakcyjne⁶. Jedną z największych zalet diagramów klas UML jest możliwość opisywania związków, które między nimi występują. Jeśli chodzi o relacje pomiędzy klasami, mogą w nich występować cechy krotności⁷:

- 1 – oznacza jeden obiekt,
- 0-3 – od zera do trzech obiektów,
- „*?” – dowolną ilość obiektów,
- 3 - * – od trzech do dowolnej ilości obiektów.

Co ważne, krotności należy umieszczać po obu stronach zależności. W przypadku gdy nie zostanie podana krotność, trzeba przyjąć, że jej wartość wynosi 1. Związki między klasami możemy podzielić na pięć typów takich jak zależność, asocjacja, agregacja częściowa, agregacja całkowita i dziedziczenie. Zależność między klasami informuje, że jedna z nich musi mieć informacje o drugiej, aby korzystać z jej obiektów. Występuje w momencie, w którym zmiana specyfikacji jednej klasy powoduje konieczność wprowadzenia zmian w innej klasie. Wynika z tego, że obie klasy są od siebie zależne. Jeśli chodzi o oznaczenia, używa się trzech typów:

- `<<call>>` – operacje w klasie A wywołują opera-

- cje w klasie B,
- `<<create>>` – klasa A tworzy instancje w klasie B,
- `<<instantitate>>` – obiekt A jest instancją klasy B,
- `<<use>>` – do zaimplementowania klasy A wymagana jest klasa B.

Pierwszą i jednocześnie najsłabszą relacją jest zależność⁸. Oznacza ona, że jedna z klas chwilowo wykorzystuje inną klasę tudzież wie o jej istnieniu. Jeśli zajdzie jakaś zmiana w jednej z klas, może to spowodować (ale nie musi) konieczność zmian w drugiej klasie. Ważne jest, żeby zaprojektowany diagram zawierał jak najmniejszą liczbę zależności, ponieważ utrudniają one rozbudowę projektu.



Rys. 1. Zastosowanie diagramu zależności

Asocjacja przedstawia czasowe powiązania pomiędzy obiektami dwóch różnych klas⁹. W tym przypadku wszystkie obiekty są od siebie niezależne, a sama asocjacja jest też używana jako alternatywny sposób zapisu cech klasy. Asocjacja jest silniejszą relacją niż opisywana wcześniej zależność. Wskazuje, że dany obiekt jest powiązany z innym przez pewien czas (czas istnienia obu obiektów nie jest jednak od siebie zależny). Jeśli w jednym z nich zajdą modyfikacje w zmiennych lub zostanie on całkowicie usunięty, drugi będzie istniał w niezmienionej formie. Co ważne, w przypadku asocjacji żaden obiekt nie jest właścicielem drugiego, a zatem, dla przykładu, nie tworzy go ani nie zarządza nim. Należy też pamiętać, że obiekty, które zostały powiązane ze sobą asocjacją, mogą mieć nazwy, które zazwyczaj występują w postaci czasownika. Asocjacja może być jednokierunkowa oraz dwukierunkowa. Nazwy asocjacji wskazują bezpośrednio na czynności, które zachodzą pomiędzy klasami.



Rys. 2. Zastosowanie diagramu asocjacji

Kolejnym typem związków pomiędzy klasami w UML jest agregacja, która stanowi silniejszą odmianę zwykłej asocjacji. Agregacja występuje w dwóch odmianach: częściowej oraz całkowitej, inaczej nazywanej kompozycją. Agregacja częściowa jest związkiem dwóch klas w formie relacji całość-część. Warto dodać, że usunięcie

5. *Diagramy klas UML*, <https://www.p-programowanie.pl/uml/diagramy-klas-uml>, dostępny w internecie [dostęp 4.12.2021].

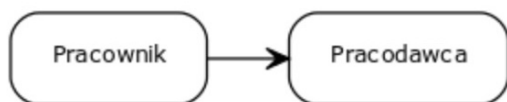
6. *UML Class Diagram Tutorial: Abstract Class with Examples*, <https://www.guru99.com/uml-class-diagram.html>, dostępny w internecie [dostęp 4.12.2021].

7. *Diagramy klas UML*, <https://www.p-programowanie.pl/uml/diagramy-klas-uml>, dostępny w internecie [dostęp 4.12.2021].

8. *UML Class Diagram Tutorial*, <https://www.lucidchart.com/pages/uml-class-diagram>, dostępny w internecie [dostęp 4.12.2021].

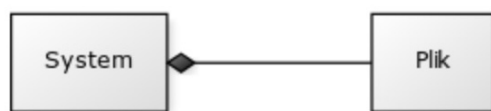
9. *UML Class Diagram Tutorial: Abstract Class with Examples*, <https://www.guru99.com/uml-class-diagram.html>, dostępny w internecie [dostęp 4.12.2021].

klasy „wszystko” nie wpłynie na klasę „fragment”. W agregacji częściowej element częściowy może należeć do elementu głównego, lecz nie jest od niego zależny. Co za tym idzie, usunięcie elementu głównego nie wpływa w żadnym stopniu na element częściowy. Dodatkowo ten ostatni nie jest przywiązany tylko do jednego elementu głównego, ponieważ może należeć do kilku z nich.



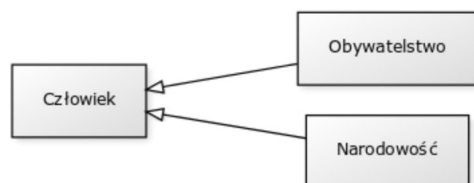
Rys. 3. Zastosowanie diagramu agregacji częściowej

Agregacja całkowita zwaną kompozycją działa podobnie do agregacji częściowej¹⁰. Największą różnicą jest fakt, że kontroluje cykl życia danej części. Oznacza to, że jeśli zostanie usunięta klasa główna, razem z nią zniknie powiązana z nią klasa częściowa. Co więcej, ani całość, ani część nie mogą bez siebie istnieć, dlatego czas ich działania jest ściśle ze sobą powiązany i się pokrywa.



Rys. 4. Zastosowanie diagramu agregacji całkowitej

Uogólnienie lub inaczej dziedziczenie tworzy hierarchię klas od tych ogólnych do najbardziej szczegółowych. Jest to główny filar paradygmatu programowania obiektowego. Umożliwia on wyodrębnienie cech wspólnych dla kilku klas i zamknięcie ich w klasie ogólnej, która ma wyższy poziom abstrakcji. Co ważne, klasy dziedziczące po klasie bazowej przyjmują jej cechy. Pozwala to na znaczne skrócenie kodu i zorganizowanie go od strony logicznej. Dziedziczenie ma kilka trybów: *overlapping*, *disjoint*, *complete* oraz *incomplete*.



Rys. 5. Zastosowanie diagramu dziedziczenia

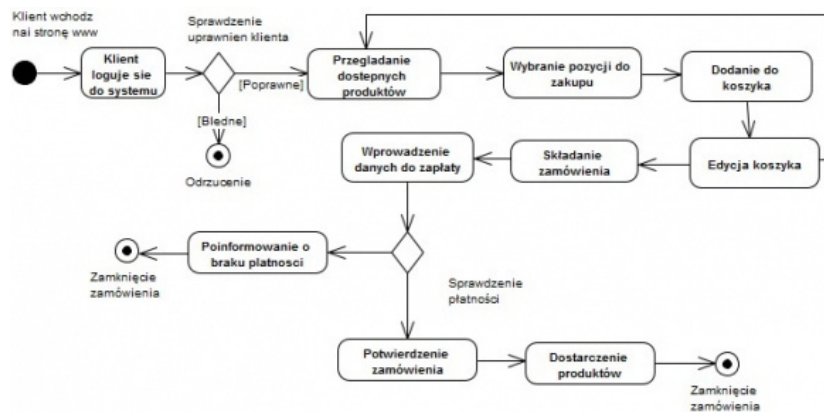
2.2. UML w perspektywie graficznej

Za pomocą języka UML w perspektywie graficznej da się przedstawić zasady działania pewnych procesów w biznesie. Dzięki temu w prosty i przejrzysty sposób na spotkaniu z klientem można zaprezentować mu sposób działania tworzonej aplikacji¹¹. Modele, które są zapisane

w języku UML, ułatwiają też rozszyfrowanie sposobu działania systemów osobom, które ten system mają tworzyć. Grupy projektowe omawiają modele i dzięki temu łatwiej jest im zrozumieć sposób, w jaki ma działać przyszła aplikacja. Graficzne modele są ustandaryzowane, więc modele tworzone w firmie A będą też rozumiane, gdy zmienimy pracę i zobaczymy inny model w firmie B. Język, którym się posługujemy, by zobrazować rzeczywistość, byłby zbyt trudny, żeby opisać działanie całej aplikacji, dlatego też stosowanie języka UML we wczesnej fazie projektu jest bardzo ważne. Na rynku funkcjonuje wiele aplikacji oferujących tworzenie modeli w języku UML. Każdy architekt oprogramowania lub programista zaczyna tworzenie oprogramowania od rozrysowania wszystkiego na kartce – nawet taki projekt na kartce można nazwać diagramem UML, jeśli jest on prawidłowo wykonany. Osoby, które znają się na języku UML, patrząc na sam model, są w stanie tylko na jego podstawie stworzyć całą aplikację. Pokazuje to, jak bardzo taki model jest w stanie ułatwić prace nad nimi.

W perspektywie graficznej języka UML można wydzielić kilka rodzajów diagramów:

- Diagram przypadków użycia,
- Diagram pakietów,
- Diagram klas,
- Diagram aktywności,
- Diagram sekwencji,
- Diagram komponentów.



Rys. 6. Przykładowy diagram UML pokazujący działanie strony www do obsługi zamówień

2.3. Diagram przypadków użycia (use case diagram).

Ten rodzaj diagramu służy do przedstawienia funkcjonalności systemu wraz z jego otoczeniem, związków zachodzących między użytkownikami a systemem i zobrazowania usług, które są widoczne z zewnątrz systemu.

Diagramy przypadków użycia zbudowane są z kilku elementów, ale odgrywają najważniejszą rolę w procesie projektowania systemu. Opisują one wymagania funkcjonalne, jakie system musi spełnić, oraz otoczenie, w którym się znajduje. W tym samym czasie wspo-

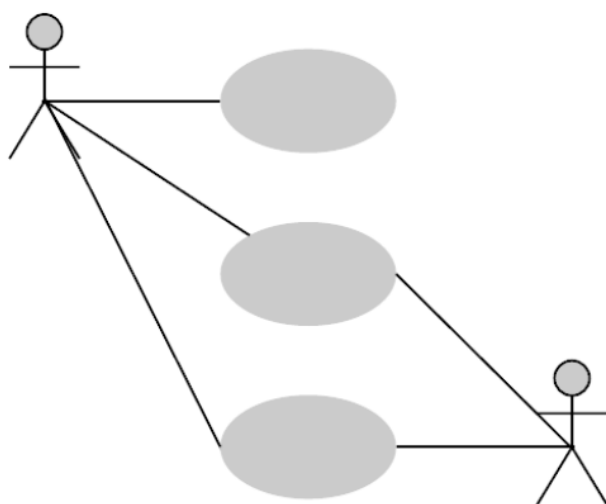
10. *Diagramy klas UML*, <https://www.p-programowanie.pl/uml/diagramy-klas-uml>, dostępny w internecie [dostęp 4.12.2021].

11. *UML – Diagramy behawioralne*, <https://it.pwn.pl/Artykuly/Zarzadzanie-projektami/UML-Diagramy-behawioralne>, dostępny w internecie [dostęp 4.12.2021].

magają komunikację między uczestnikami projektu¹². Przypadek użycia to scenariusze powiązane ze sobą wspólnym celem użytkownika. Pozwalają one na zdefiniowanie przyszłego, spodziewanego zachowania systemu, są stosowane w całej analizie systemu i mają za zadanie dostarczyć wyniki, z których użytkownik będzie mógł skorzystać. Przypadek użycia musi być powiązany z interakcją chociaż jednym „aktorem” (rolą, którą pełni użytkownik w stosunku do systemu oraz przypadków użycia) Każdy z nich możemy opisać za pomocą następujących cech:

- nazwa
- opis
- przepływ zdarzeń
- zależności i relacje
- diagramy aktywności
- wymagania specjalne
- warunki wstępne
- warunki końcowe

Najważniejszym aspektem opisującym przypadek użycia jest przepływ zdarzeń – scenariusze, które wskazują zestaw czy sekwencję kolejno wykonywanych czynności służących do zrealizowania funkcjonalności zobrazonej przez dany przypadek użycia.

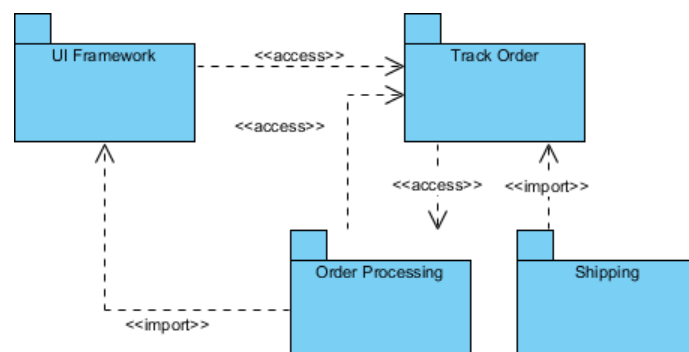


Rys. 7. Przykładowy diagram przypadków użycia

2.4. Diagram pakietów

Jest to strukturalny diagram prezentujący pakiety i relacje, które między nimi zachodzą. Służy on do modelowania agregatów bytów, jakimi są pakiety. Pozwala on na modelowanie systemu na wysokim stopniu abstrakcji, gdyż pakiety reprezentują ogromną liczbę klas, interfejsów diagramów i innych bytów – pozwala on na przedstawienie tylko podstawowych cech systemu. Pakiet to uniwersalny mechanizm służący do organizowania elementów w grupy.

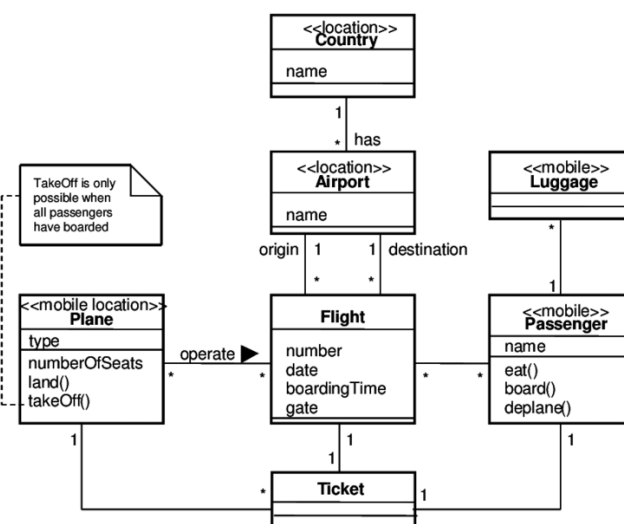
Z jego pomocą grupuje się elementy modelu i diagramy. Pakiety to jeden z najczęściej stosowanych elementów, dzięki którym utrzymuje się porządek w repozytorium projektu¹³.



Rys. 8. Zastosowanie diagramu pakietów

2.5. Diagram klas

Ten rodzaj diagramu obejmuje pewien zbiór klas, interfejsów i kooperacji oraz związki między nimi¹⁴. Stanowi on opis siatki systemu, który uwypukla związki między klasami, pomijając pozostałe charakterystyki. Podczas modelowania złożonych systemów nie trzeba przedstawiać ich struktury na jednym diagramie. Złożenie wszystkich diagramów (ich elementów i relacji) stanowi kompletny model. Podzbiory zbiorów klas użyte na diagramach klas są wybierane celowo i stanowią wynik decyzji analitycznych i projektowych.



Rys. 9. Zastosowanie diagramu klas

2.6. Diagram aktywności

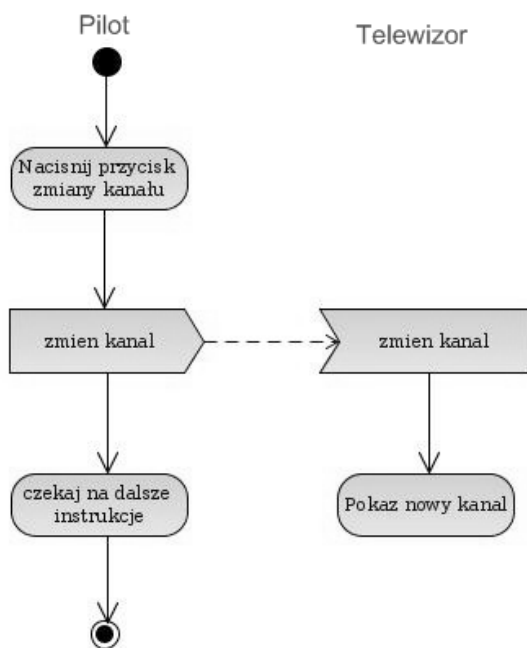
Diagram ten, zwany również diagramem interakcji, służy do modelowania dynamicznych aspektów systemu¹⁵. Jego główną funkcją jest przedstawienie sekwencji kroków, które są wykonywane przez modelowany fragment

12. *UML Use Case Diagram Tutorial*, <https://www.lucidchart.com/pages/uml-use-case-diagram>, dostępny w internecie [dostęp 4.12.2021]

13. *What is Package Diagram?*, <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-package-diagram/>, dostępny w internecie [dostęp 4.12.2021].

15. *Diagram aktywności*, <https://wolski.pro/diagramy-uml/diagram-aktywnosci/>, dostępny w internecie [dostęp 4.12.2021].

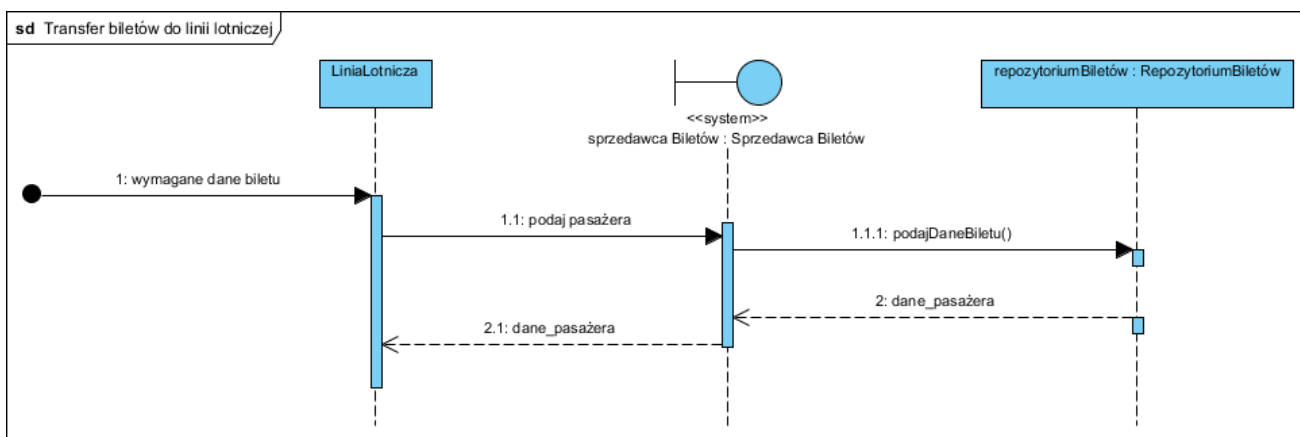
systemu.



Rys. 10. Zastosowanie diagramu aktywności

2.7. Diagram sekwencji

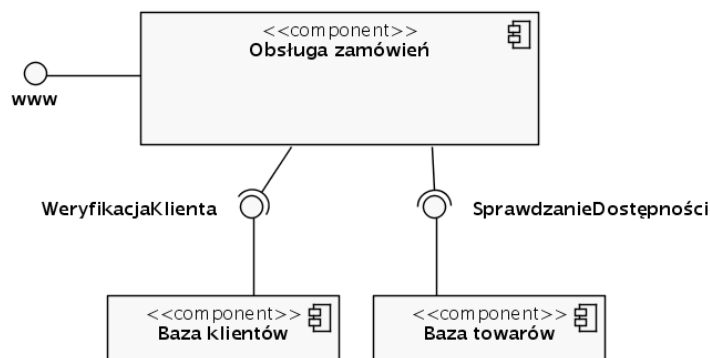
Ów diagram służy do prezentowania interakcji pomiędzy obiektami, ale z uwzględnieniem w czasie komunikatów, jakie są między nimi przesyłane¹⁶. Tego rodzaju diagramy pozwalają uzyskać odpowiedź na pytanie, jak przebiega komunikacja pomiędzy obiektami w czasie. Stanowią również jedną z podstawowych technik zachowania systemu składających się na realizację przypadku użycia.



Rys. 11. Zastosowanie diagramu sekwencji

2.8. Diagram komponentów

Diagram ten ilustruje organizację i zależności między komponentami¹⁷. Prezentuje on system na wyższym poziomie abstrakcji niż diagram klas. Służy do określania szczegółów niezbędnych do budowy systemów.



Rys. 12. Zastosowanie diagramu komponentów

2.9. Język UML w praktyce

Na rynku jest wiele aplikacji oferujących tworzenie modeli w języku UML¹⁸. Każdy architekt oprogramowania lub programista tworzenie aplikacji zaczyna od rozrysowania wszystkiego na kartce – nawet taki projekt na kartce można nazwać diagramem UML, jeśli jest on prawidłowo wykonany. Można znaleźć wiele darmowych programów, które oferują tworzenie diagramów UML graficznie, oprócz tego znajdziemy również w nich gotowe rozwiązania. Wybierając ogólnie przygotowany szablon możemy być pewni, że taki model jest dobrze wykonany, wystarczy tylko dostosować go pod tworzoną aplikację. Według założeń projektowanie aplikacji rozpoczyna się od rozmowy podmiotu zamawiającego program z analitykiem, który próbuje zrozumieć specyfikę i potrzeby klienta. Analityk przekazuje informacje architektowi aplikacji i to właśnie zazwyczaj on odpowiedzialny jest za zaprojektowanie całego modelu. Gdy architekt stworzy taki model, trafia

on do programistów, których zadaniem jest przełożyć model graficzny na gotową aplikację. Architekt aplikacji to zazwyczaj osoba z ogromnym doświadczeniem. Potrafi on zrozumieć aplikację jako całość. Aplikacja czy projektowany system to nie tylko zbiór kodu, to też połączenia z bazami danych, integracje różnych systemów. Architekt oprogramowania cały czas bierze udział w pracach zespołu, programiści mogą coś zmieniać w projekcie, pojawiają się nowe pomysły i korekty. Oprócz tego czuwa

16. *Diagram sekwencji*, <https://wolski.pro/diagramy-uml/diagram-sekwencji/>, dostępny w internecie [dostęp 4.12.2021].

17. *Diagram komponentów (component diagram)*, <http://zasoby.open.agh.edu.pl/~09sbfrazek/diagram-komponentow%2C1%2C17.html>, dostępny w internecie [dostęp 4.12.2021].

18. *Best UML Tools*, <https://www.guru99.com/best-uml-tools.html>, dostępny w internecie [dostęp 4.12.2021].

on nad tym czy to, co zaprojektował, jest zgodne z tym, co piszą programiści¹⁹.

3. Podsumowanie

Modele UML tworzy się po to, by zobrazować, jak ma działać program. Pozwala to lepiej zrozumieć potrzeby klienta i przedstawić działanie programu. Dzięki ujednoliconemu językowi da się postrzegać system z różnych punktów widzenia. Najważniejszym elementem podczas tworzenia oprogramowania jest komunikacja. Przełożenie myśli klienta na język informatyki to bardzo trudne działanie. Język, którym posługujemy się na co dzień, okazał się zbyt skomplikowany, by opisywać nim cały system. Powstanie UML umożliwiło w łatwy sposób opisywanie zasad funkcjonowania tworzonej aplikacji, pokazanie systemu z różnych punktów widzenia. Modele zapisane w tym języku prezentują system w sposób graficzny w postaci diagramów. Pokazują aplikację od ogółu do szczegółu, umożliwiając oglądanie modelu systemu z wybraną w danym momencie szczegółowością.

W dzisiejszych czasach aplikacje muszą być zaprojektowane tak, by w przyszłości można było je rozszerzać o nowe funkcjonalności. Model UML pozwala zaprojektować je już w pierwszych fazach tworzenia projektu. Co więcej, programiści nie muszą implementować od razu wszystkich dodatków. Mogą je dodawać wtedy, gdy zajdzie taka potrzeba. Modelowanie złożonych systemów jest trudnym zadaniem i angażuje się w to wiele osób o różnym sposobie postrzegania systemu. Modele w języku UML przedstawiają zazwyczaj „4+1” perspektywy. Cztery pierwsze to obraz wewnętrznej struktury programu na różnych stopniach abstrakcji i szczegółowości. Ostatnia perspektywa prezentuje funkcjonalność systemu widzianą przez jednego z użytkowników.

1. Perspektywa przypadków użycia – opisuje funkcjonalność aplikacji
2. Perspektywa logiczna – opisuje sposób realizacji funkcjonalności
3. Perspektywa implementacyjna – opisuje poszczególne moduły
4. Perspektywa procesowa – opisuje właściwości pozafunkcjonalne aplikacji
5. Perspektywa wdrożenia – służy integratorom i instalatorom systemu.

Modele w języku UML ułatwiają pracę programistom, osobom zaangażowanym w stworzenie danej aplikacji. Dzięki nim można też zaprezentować klientowi sposób działania systemu. W łatwy sposób można wytłumaczyć, jak aplikacja będzie na przykład odbierać zamówienia. Model w języku UML jest niezbędny przy większych projektach – dzięki niemu programiści mogą się dowie-

dzieć, w jaki sposób mają coś zaprogramować i jak coś ma działać. UML nie jest językiem programowania graficznego, jednak modele w nim zapisane mogą być wprost powiązane z wieloma językami programowania. Model utworzony w języku UML można przekształcić, np. w tabelę relacyjnej bazy danych.

Bibliografia

1. *Best UML Tools*, <https://www.guru99.com/best-uml-tools.html>, dostępny w internecie [dostęp 4.12.2021]
2. *Diagram aktywności*, <https://wolski.pro/diagramy-uml/diagram-aktywnosci/>, dostępny w internecie [dostęp 4.12.2021]
3. *Diagram sekwencji*, <https://wolski.pro/diagramy-uml/diagram-sekwencji/>, dostępny w internecie [dostęp 4.12.2021].
4. *Diagram komponentów (component diagram)*, <http://zasoby.open.agh.edu.pl/~09sbfraczek/diagram-komponentow%2C1%2C17.html>, dostępny w internecie [dostęp 4.12.2021]
5. *Diagramy klas UML*, <https://www.p-programowanie.pl/uml/diagramy-klas-uml>, dostępny w internecie [dostęp 4.12.2021]
6. *Poradnik tworzenia diagramów i modelowania bazy danych w UML*, <https://www.microsoft.com/pl-pl/microsoft-365/business-insights-ideas/resources/guide-to-uml-diagramming-and-database-modeling>, dostępny w internecie [dostęp 4.12.2021].
7. *UML*, <https://mfiles.pl/pl/index.php/UML>, dostępny w internecie [dostęp 4.12.2021]
8. *UML Class Diagram Tutorial*, <https://www.lucidchart.com/pages/uml-class-diagram>, dostępny w internecie [dostęp 4.12.2021]
9. *UML Class Diagram Tutorial*, <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/uml-class-diagram-tutorial/>, dostępny w internecie [dostęp 4.12.2021]
10. *UML Class Diagram Tutorial: Abstract Class with Examples*, <https://www.guru99.com/uml-class-diagram.html>, dostępny w internecie [dostęp 4.12.2021]
11. *UML – Diagramy behawioralne*, <https://it.pwn.pl/Artykuly/Zarzadzanie-projektami/UML-Diagramy-behawioralne>, dostępny w internecie [dostęp 4.12.2021]

¹⁹. *Poradnik tworzenia diagramów i modelowania bazy danych w UML*, <https://www.microsoft.com/pl-pl/microsoft-365/business-insights-ideas/resources/guide-to-uml-diagramming-and-database-modeling>, dostępny w internecie [dostęp 4.12.2021]

12. *UML Use Case Diagram Tutorial*, <https://www.lucid-chart.com/pages/uml-use-case-diagram>, dostępny w internecie [dostęp 4.12.2021]

13. *Unified Modeling Language*, <https://it.pwn.pl/Artykuly/Zarzadzanie-projektami/Unified-Modeling-Language-UML>, dostępny w internecie [dostęp 4.12.2021]

14. *What is Package Diagram?*, <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-package-diagram/>, dostępny w internecie [dostęp 4.12.2021]



Zezwala się na korzystanie z *Wprowadzenie do modelowania w języku UML* na warunkach licencji Creative Commons Uznanie autorstwa 4.0 (znanej również jako CC-BY), dostępnej pod adresem <https://creativecommons.org/licenses/by/4.0/pl/> lub innej wersji językowej tej licencji lub którejkolwiek późniejszej wersji tej licencji, opublikowanej przez organizację Creative Commons.