

УДК 004

doi:10.20998/2413-4295.2022.02.08

РОЗРОБЛЕННЯ АВТОМАТИЗОВАНОЇ БІБЛІОГРАФІЧНОЇ СИСТЕМИ

С. О. ЦИБУЛЬНИК*, Д. С. БІДНИК, Д. О. ПІВТОРАК

кафедра комп'ютерно-інтегрованих оптичних та навігаційних систем, Національний технічний університет України «Київський політехнічний інститут ім. Ігоря Сікорського», Київ, УКРАЇНА

*e-mail: tsybulnik.s.a@gmail.com

АНОТАЦІЯ В Україні більшість наукових установ і закладів вищої освіти фінансуються державою. Майже всі наукові дослідження проводяться переважно в державному секторі, тоді як приватні університети та компанії зазвичай не проводять жодних досліджень самостійно. Державні заклади вищої освіти фінансуються за кошти державного бюджету, і брак коштів є одним з основних факторів, які заважають впроваджувати нові інформаційні технології та системи в повсякденну рутину. Аналіз наукових досягнень з використанням бібліографічної інформації став значною частиною науково-технічного менеджменту в багатьох країнах світу, але на відміну від Словенії та Польщі, Україна не має єдиної бібліографічної бази даних та відповідних пошукових інтерфейсів. Тому запропоновано розробити автоматизовану бібліографічну систему, що спростить процеси створення, редагування, пошуку, сортування, статистики, видалення бібліографічної інформації науково-педагогічного працівника та дасть змогу автоматизувати формування персоналізованих звітів. Для реалізації запропонованої автоматизованої системи проведено огляд існуючих баз бібліографічної інформації та пошукових інтерфейсів. Зроблено висновок, що вони не можуть забезпечити науковців України необхідними функціональними можливостями для формування бібліографічних описів власних наукових робіт. Для проектування архітектури мобільного додатку на базі операційної системи Android обрано клас багаторівневої архітектури та шаблон MVVM. Спроектовано архітектуру, яка містить три рівні та п'ять шарів, на яких розміщуються окремі підсистеми. Для тестування додатку на різних розмірах екранів, а також на різних версіях програмного інтерфейсу був використаний емулятор Pixel та смартфон Xiaomi Redmi 5 Plus. Реалізовано оформлення введених даних за двома стандартами, а саме: ДСТУ 8302-2015 та IEEE. Реалізовано алгоритми пошуку та сортування записів за необхідними критеріями. Реалізовано збереження списку усіх необхідних бібліографічних описів у текстовому форматі. Додатково реалізовано можливість його відправки на електронну пошту. У подальшому планується розширити список стандартів оформлення бібліографічної інформації та удосконалити алгоритми пошуку і сортування.

Ключові слова: бібліографічна база даних; Android; Java; програмна архітектура; програмне забезпечення; MVVM

DEVELOPMENT OF AN AUTOMATED BIBLIOGRAPHIC SYSTEM

S. TSYBULNYK, D. BIDNYK, D. PIVTORAK

Department of Computer-Integrated Optical and Navigation Systems, Igor Sikorsky Kyiv Polytechnic Institute, Kyiv, UKRAINE

ABSTRACT Most research and higher education institutions of Ukraine are funded by the state. Almost all research is conducted mainly in the public sector, while private universities and companies usually do not conduct any research. Public higher education institutions are financed from the state budget and lack of funds is one of the main factors hindering the introduction of new information technologies and systems into the daily routine. Analysis of scientific achievements using bibliographic information has become a significant part of scientific and technical management in many countries around the world, but unlike Slovenia and Poland, Ukraine does not have a single bibliographic database and relevant search interfaces. Therefore, the aim of the work is to develop an automated bibliographic system that will simplify the process of creating, editing, searching, sorting, statistics, deleting bibliographic information of research and teaching staff and will automate the formation of personalized reports. To implement the proposed automated system, a review of existing databases of bibliographic information and search interfaces was conducted. It is concluded that they cannot provide Ukrainian scientists with the necessary functionality to form bibliographic descriptions of their own research papers. To design the architecture of a mobile application based on the Android operating system, a multi-level architecture class and an MVVM template were selected. Designed architecture contains three levels and five layers on which separate subsystems are located. The Pixel emulator and Xiaomi Redmi 5 Plus smartphone were used to test the application on different screen sizes, as well as on different versions of the software interface. The application was developed in the AndroidStudio software environment. The design of the entered data according to two standards was implemented, namely: DSTU 8302-2015 and IEEE. Algorithms for searching and sorting records according to the necessary criteria are implemented. Saving the list of all necessary bibliographic descriptions in text format is implemented. Additionally, the possibility of sending it to e-mail is implemented. In the future it is planned to expand the list of standards for bibliographic information and improve search and sorting algorithms.

Keywords: bibliographic database; Android; Java; software architecture; software; MVVM

Вступ

Важливим інструментом для розвитку будь-якого професіонала, особливо для тих, хто працює у

галузі освіти та науки, є інформація, тобто пов'язані між собою концептуально дані або відомості, які можуть при їх використанні змінювати уявлення про об'єкти чи явища. Вдале застосування актуальної

інформації дозволяє просуватися не лише в академічній та науковій сферах, але й у повсякденному житті. Проте на сьогоднішній день є доступним величезний обсяг інформації, що проявляється у наявності сотень тисяч фахових підручників та мільйонів статей у періодичній літературі. Саме тому розроблено велику кількість автоматизованих (або автоматичних) інформаційних систем, які дозволяють зберігати, оброблювати та систематизувати подібну інформацію.

Проте навіть автоматизовані системи не позбавлені недоліків. Вони також, як і люди, можуть використовувати абстрактну, штучну і помилкову інформацію, яка часто не має об'єктивного чи адекватного відображення ні в природі, ні в суспільстві. Тому визначальним фактором у дослідженні шляхів вирішення будь-якої проблеми, безсумнівно, є розроблення відповідної стратегії отримання адекватної інформації.

Ще в 1960-х роках була заснована концепція універсального бібліографічного контролю [1], яка використовувала узгоджені на міжнародному рівні принципи каталогізації. Універсальний підхід до формування бібліографічної інформації є одним з основних методів забезпечення адекватного пошуку наукових даних у всьому світі.

В ідеалі, бібліографічна інформація має бути універсальною, загальнодоступною та містити основні бібліографічні дані про всі видання у всіх країнах [2], але на практиці цього досягти досі не вдалося. Для проведення оптимального бібліографічного пошуку важливо розуміти різні етапи, які в нього входять, знати, як реалізувати вдалу стратегію пошуку, мати уявлення про доступні бази даних, а також вміти користуватися бібліографічними менеджерами.

Бібліографічний пошук – це серія дій, спрямованих на пошук документів, які стосуються певної теми та покликані оптимізувати стан знань, які сприймаються як неповні або не відповідають певним вимогам [3]. Бібліографічний пошук є початковою фазою будь-якої наукової роботи та спрямований на збір достатньої кількості інформації щодо актуальності задачі, яка вирішується. Зазвичай це неструктурований процес, який містить також і певний рівень стохастичності. Неструктурований бібліографічний пошук легко може стати трудомістким і нескінченним процесом.

У зв'язку з постійно зростаючим обсягом наукової літератури підвищення ефективності пошуку бібліографічних даних стало актуальним завданням. Тому, щоб максимізувати ефективність результату бібліографічного пошуку при обмежених часі та ресурсах, науковці розробляють нові підходи до його проведення, які включають в себе наступні: формування стратегій пошуку [3,4]; виявлення та структурування інформації за рахунок формування її функціональних ознак [5]; нейронні мережі [6];

технології семантичних та пов'язаних даних [7]; графі [8-10], тощо.

Загалом, пошук здійснюється у певній бібліографічній базі даних, яка містить посилання на опубліковану літературу та може включати багато різних статей, книг, законів, тощо [3]. Інформація, яка міститься у таких базах даних, структурована та впорядкована в записи та поля, індексована дескрипторами та може включати посилання на повний текст. На сьогодні існує велика кількість бібліографічних баз даних, а також розробляються нові для задоволення вимог певних галузей науки (наприклад, [11-15]).

Проте, як показує практика [16], пошук навіть у декількох базах даних з використанням пошукових інтерфейсів не забезпечує повного і якісного результату. Бібліографічний менеджер (пошукова система, пошуковий інтерфейс) – це програмне забезпечення, яке керує пошуком бібліографічної інформації у відповідних базах даних [3,17,18].

В Україні, як і в Польщі [19], більшість наукових установ і закладів вищої освіти фінансуються державою. Майже всі наукові дослідження проводяться переважно в державному секторі, тоді як приватні університети та компанії зазвичай не проводять жодних досліджень самостійно. Державні заклади вищої освіти фінансуються за кошти державного бюджету, і брак коштів є одним з основних факторів, які заважають впроваджувати нові інформаційні технології та системи в повсякденну рутину.

На сьогодні інформаційні системи добре охоплюють лише найважливіші сфери діяльності університету, наприклад, управління персоналом, документообігом або оцінювання студентів. Зазвичай ці рішення навіть в рамках одного університету надаються різними компаніями або розробляються власними силами, що дуже сильно впливає на кінцеву якість такого програмного забезпечення. При цьому, як правило, ігноруються або відводяться на задній план потреби науково-педагогічного персоналу в запровадженні автоматизованої системи для збереження, сортування та надання персональної бібліографічної інформації.

Аналіз наукових досягнень з використанням бібліографічної інформації став значною частиною науково-технічного менеджменту в багатьох країнах світу. Наприклад, бібліографічна інформація дослідників у Словенії [20] на національному рівні включає оцінку бібліографічних показників наукової діяльності дослідників, дослідницьких груп, проектів, програм та організацій. На відміну від Словенії та Польщі [19,20], Україна не має єдиної бібліографічної бази даних та відповідних пошукових інтерфейсів. Саме тому щороку тисячі викладачів в Україні змушені оформлювати у відповідності до певного стандарту перелік своїх наукових праць у вигляді набору бібліографічних описів для: проходження конкурсу на заміщення посади; захисту дисертації

кандидата або доктора наук; участі в різноманітних конкурсах та наукових роботах чи проектах (зокрема державних); отримання наукового звання; формування щорічних звітів структурних підрозділів з науки; тощо [2]. Формування такого списку відбувається, як правило, в ручному режимі і потребує багато часу на обробку та аналіз бібліографічної інформації, оскільки для різних потреб наповнення списку наукових праць буде відрізнятися: статті у періодичних виданнях за останні 5 років; усі публікації за весь період діяльності; тільки документи, які підтверджують наявність авторських прав; тощо. Кількість різноманітних звітів за персональними науковими показниками може коливатися від одного до 7-8 за рік, що призводить до значних витрат часу.

Враховуючи усі названі вище недоліки, пропонується розробити автоматизовану бібліографічну систему, яка спростить процеси створення, редагування, пошуку, сортування, статистики, видалення (та інші) бібліографічної інформації (бібліографічних описів) науково-педагогічного працівника та дасть змогу автоматизувати формування персоналізованих звітів.

Мета роботи

Метою роботи є розроблення автоматизованої бібліографічної системи у вигляді мобільного додатку на базі операційної системи Android для збереження та організації персональних бібліографічних показників.

Архітектура системи

З моменту свого утворення область комп'ютерних наук зіткнулася з проблемами, що були пов'язані зі складністю програмних систем. Раніше ці проблеми вирішувалися безпосередньо розробниками шляхом правильного вибору структур даних, алгоритмів та розмежування повноважень. Сьогодні ж зниження складності системи досягається за рахунок проектування якісної програмної архітектури. Під програмною архітектурою мається на увазі структура системи, принципи взаємодії її компонентів, а також аналіз і реалізація її властивостей.

Архітектура програмного забезпечення має декілька загальних класів, які є аналогічними різним типам креслень в будівництві споруд. Наприклад, клас багаторівневої (шаруватої, багат шарової) архітектури програмного забезпечення сьогодні є найбільш часто використовуваним. Метою шаруватої архітектури є організація окремих компонентів в горизонтальні логічні шари та фізичні рівні.

У закритій багаторівневій архітектурі шар може викликати лише шар, який знаходиться безпосередньо під ним. На рис. 1 наведено діаграму закритої багаторівневої архітектури.

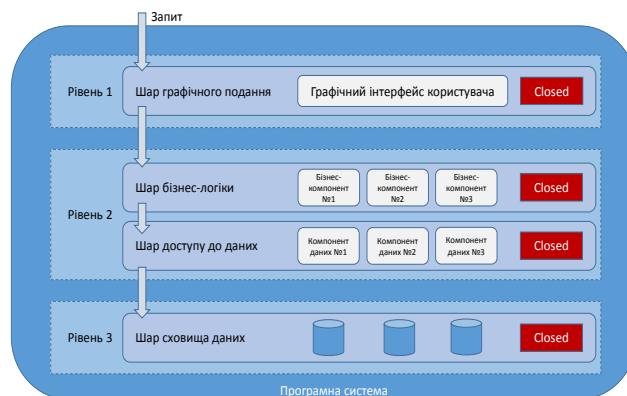


Рис. 1 – Приклад закритої багаторівневої архітектури

На наведеній діаграмі кожен шар позначений як закритий. Це означає, що для того, щоб запит пройшов до самого нижнього шару, він має пройти послідовно через кожен шар. Наприклад, коли запит ініціюється через рівень графічного подання, він спочатку буде проходити через рівень бізнес-логіки, потім через шар доступу до даних і, нарешті, дійде до шару сховища даних.

Метою закритої багаторівневої архітектури є забезпечення повної ізоляції шарів один від одного. Це дасть змогу стверджувати, що внесені в один шар архітектури зміни, не впливатимуть на інші шари, тобто реалізується принцип низького зчеплення.

Для реалізації даного класу архітектури було обрано архітектурну модель (шаблон) MVVM [2]. MVVM (Model-View-ViewModel) є однією з можливих реалізацій моделі абстрактної машини. Використання даного шаблону полягає у відокремленні логіки додатку від його візуальної частини та поділу програмної системи на три рівні (рис. 2): модель (Model), модель подання (ViewModel) і уявлення або вигляд (View).

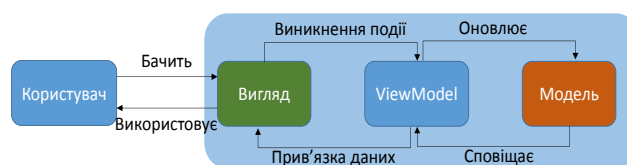


Рис. 2 – Шаблон MVVM

Серед переваг шаблону MVVM можна виокремити наступні:

- низьке зчеплення між ViewModel і View;
- забезпечує найкращу сумісність з керованим тестами розробленням програмного забезпечення.

Його недоліки:

- має великий розмір коду;

- кожен елемент інтерфейсу користувача потребує компоненти, які будуть викликати події.

На основі закритої багаторівневої архітектури та шаблону MVVM було спроектовано архітектуру автоматизованої бібліографічної системи (рис. 3). Дана архітектура містить три рівні та п'ять шарів, які включають одну чи більше підсистем кожен. У порівнянні з класичним уявленням багаторівневої архітектури (рис. 1) у даному випадку додано четвертий шар – шар програмної логіки, який знаходиться на третьому рівні і містить підсистеми обробки даних та транспортування. За своєю логікою дані підсистеми являються низькорівневими та не можуть бути включеними до шару з реалізацією бізнес-правил, тому вони розміщені саме в рівні моделі.

Кожна підсистема, яка присутня в спроектованій архітектурі, у свою чергу, поділяється на окремі модулі, а саме [2]:

- підсистема користувача: новий користувач, існуючий користувач;
- підсистема формування правил: уведення даних, редагування даних, видалення даних;
- підсистема формування даних для виведення: виведення помилок, формування звітів, виведення персональних даних;
- підсистема обробки даних: пошук, сортування;
- підсистема транспортування: відправка на відображення, відправка на e-mail, збереження на внутрішньому накопичувачі.

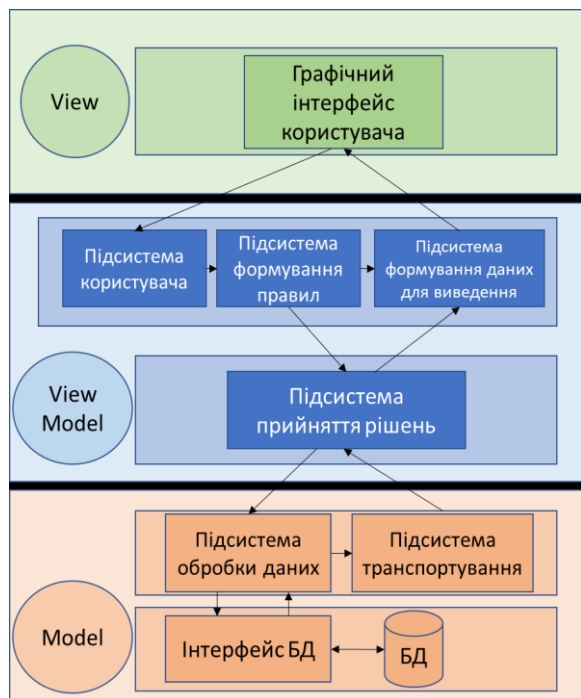


Рис. 3 – Архітектура автоматизованої бібліографічної системи

Реалізація даних підсистем дасть змогу забезпечити автоматизовану бібліографічну систему основними функціональними можливостями, необхідними для збереження персональних бібліографічних описів науково-педагогічного працівника та формування звітів з можливістю їх відправки на електронну пошту.

Результати розроблення

Після установки та налаштування Android Studio, можна також побачити, що автоматично встановилася система автоматизованої збірки додатків Gradle, яка (у відповідності до [2]) буде використовуватися в подальшому. Для початку роботи було завантажено деякі додаткові бібліотеки, а саме:

- recyclerview – оптимізує роботу зі списками, що підвищує продуктивність;
- cardview – слугує контейнером для компонентів та дозволяє змінювати елементи оболонки;
- room – спрощує роботу з базою даних SQLite;
- livedata – сховище даних, яке працює за принципом шаблону «Спостерігач». Ця бібліотека дає змогу розміщувати об'єкт у сховищі та отримувати інформацію про статус цього об'єкту.

Для демонстрації роботи додатку можна використовувати вбудований емулятор смартфона або фізичний пристрій (смартфон), який працює на операційній системі Android. Для тестування розроблюваного додатку на різних розмірах екранів, а також на різних версіях програмного інтерфейсу був використаний емулятор Pixel та смартфон Xiaomi Redmi 5 Plus.

Створення графічних компонентів додатку відбувається за допомогою використання мови розмітки XML. Усі графічні елементи розміщуються в ресурсах самого проекту, що дає можливість відокремити графічну частину від програмної логіки. Це призводить до простоти написання та тестування коду.

Для того, щоб використовувати елементи інтерфейсу в Java-коді, їм назначаються відповідні унікальні ідентифікатори в XML файлі. Елементи інтерфейсу є звичайними Java-класами і, якщо інтерфейс цього потребує, їх можна створювати не за допомогою маніпуляцій з графічними елементами, а програмно.

Додаток на основі операційної системи Android складається з набору активностей (Activity), кожна з яких відповідає окремому графічному вікну додатка. З їх допомогою користувач може взаємодіяти з графічним інтерфейсом. Життєвий цикл кожної активності являє собою сукупність станів, через які проходить кожна активність під час своєї роботи.

Враховуючи той факт, що додаток реалізовано з використанням архітектурного шаблону MVVM, усі створені активності не містять в собі програмної або бізнес-логіки, а слугують лише для відображення певної інформації та переходу між графічними вікнами. Ці класи розміщено в папці View, яка містить активності та графічні елементи, які пов'язані з графічним інтерфейсом користувача.

Вся програмна логіка розташована в папці Model і відокремлена від графічної складової додатку. Model містить всі моделі бібліографічних описів різних типів, базу даних на основі цих моделей, а також класи які оброблюють ці дані.

За операції введення, виведення даних та бізнес-логіку відповідає група класів, які знаходяться в папці ViewModel. Дані класи використовують функції бібліотеки lifecycle і «пов'язують» графічний інтерфейс з програмною логікою. Це потрібно для того, щоб дані автоматично оновлювалися при кожній їх зміні та відображалися без затримок чи втрат.

На даному етапі процесу розроблення реалізовано оформлення введених даних за двома стандартами, а саме: ДСТУ 8302-2015 та IEEE. Після проведення пошуку та сортування записів за необхідними критеріями список усіх необхідних бібліографічних описів зберігається у текстовому форматі. Також за бажанням його можна відправити на електронну пошту. Усі записи у збереженому списку в залежності від обраного стандарту мають вигляд (наприклад, для тез доповідей) або «G. R. Faulhaber, «Design of service systems with priority reservation,» in *Conf. Rec. 1995 IEEE Int. Conf. Communications*, pp. 3–8.», або «Святецька А. В. Діалектизи у повісті М. Коцюбинського «Тіні забутих предків»: семантико-стилістичний аспект. *Стратегії розвитку та пріоритетні завдання філологічних наук*: матеріали Всеукр. наук.-практ. конф., м. Запоріжжя, 19—20 жовт. 2018 р. / Класич. приват. ун-т. Запоріжжя, 2018. С. 19-23».

Приклади активностей створення бібліографічного запису наведено на рис. 4. У разі необхідності список наявних активностей можна розширити для забезпечення вимог інших стандартів оформлення бібліографічних описів.

У подальшому планується розширити список стандартів оформлення бібліографічної інформації та удосконалити алгоритми пошуку і сортування.

Висновки

Аналіз наукових досягнень наукових та науково-педагогічних працівників є одним з основних факторів оцінювання їх професійної діяльності в багатьох країнах світу. На відміну від, наприклад, Польщі та Словенії Україна не має єдиної бібліографічної бази даних та відповідних пошукових інтерфейсів, щоб забезпечити менеджмент власних бібліографічних описів.

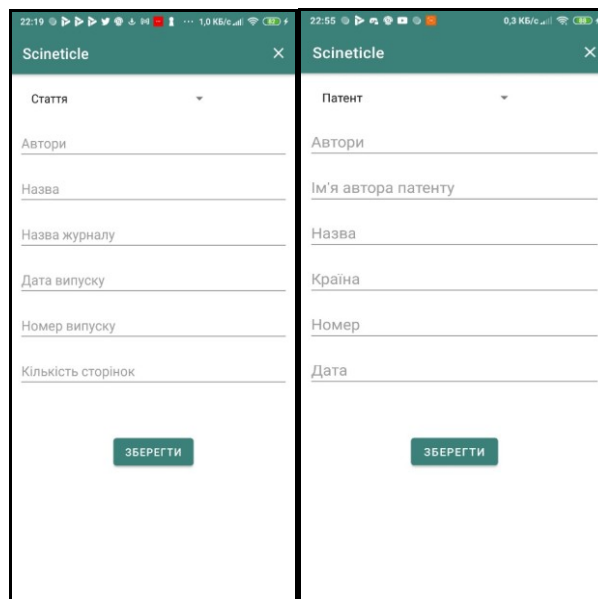


Рис. 4 – Приклади активностей створення бібліографічного опису

Саме тому на базі операційної системи Android у середовищі розроблення AndroidStudio створено автоматизовану бібліографічну систему. Її архітектуру спроектовано як багаторівневу на основі шаблону MVVM. Розроблено алгоритми бізнес-логіки, які дозволяють вносити в систему бібліографічні описи за двома стандартами, а саме: ДСТУ 8302-2015 та IEEE. Реалізовано алгоритми пошуку та сортування записів, а також формування та збереження списку за необхідними критеріями.

У подальшому планується розширити перелік доступних для оформлення стандартів, а також удосконалити алгоритми пошуку та сортування для забезпечення належного рівня якості.

Список літератури

1. Willer M., Dunsire G. *Bibliographic Information Organization in the Semantic Web*. Chandos Publishing, 2013. 318 p. doi: 10.1016/B978-1-84334-731-6.50001-6.
2. Цибульник С. О., Бідник Д. С. Проектування архітектури автоматизованої бібліографічної системи. *Вісник Національного технічного університету «ХПІ»*. Серія: Нові рішення в сучасних технологіях. Харків: НТУ «ХПІ», 2021. № 2 (8). С. 83–89. doi: 10.20998/2413-4295.2021.02.12.
3. Campos-Asensio C. How to develop a bibliographic search strategy. *Enfermería Intensiva (English ed.)*. 2018. Vol. 29, Iss. 4. P. 182–186. doi: 10.1016/j.enfie.2018.09.001.
4. Fernández-Ananín S., Rodríguez J., Soler E. Then, how do I document the idea? *Bibliographic search engines. Cirugía Española (English Edition)*. 2022. doi: 10.1016/j.cireng.2022.04.020.
5. Fayemi P.-E., Duci S., Fayolle T., Maranzana N., Bersano G. *Ideality & Bio-Inspired Based Collaborative*

- Bibliographic Search Method. *Procedia CIRP*. 2016. Vol. 39. P. 138–143. doi: 10.1016/j.procir.2016.01.179.
6. Yun J. Generalization of bibliographic coupling and co-citation using the node split network. *Journal of Informetrics*. 2022. Vol. 16. Iss. 2. doi: 10.1016/j.joi.2022.101291.
7. Rico M., Vila-Suero D., Botezan I., Gómez-Pérez A. Evaluating the impact of semantic technologies on bibliographic systems: A user-centred and comparative approach. *Journal of Web Semantics*. 2019. Vol. 59. doi: 10.1016/j.websem.2019.03.001.
8. Lee O.-J., Jeon H.-J., Jung J. Learning multi-resolution representations of research patterns in bibliographic networks. *Journal of Informetrics*. 2021. Vol. 15. Iss. 1. doi: 10.1016/j.joi.2020.101126.
9. Zhu Y., Yan E. Searching bibliographic data using graphs: A visual graph query interface. *Journal of Informetrics*. 2016. Vol. 10. Iss. 4. P. 1092–1107. doi: 10.1016/j.joi.2016.09.005.
10. Zhu Y., Yan E., Song I.-Y. A natural language interface to a graph-based bibliographic information retrieval system. *Data & Knowledge Engineering*. 2017. Vol. 111. P. 73–89. doi: 10.1016/j.datak.2017.06.006.
11. Sile L. Entanglement of bibliographic database content and data collection practices: Rethinking data integration using findings from a European study. *Procedia Computer Science*. 2019. Vol. 146. P. 201–207. doi: 10.1016/j.procs.2019.01.094.
12. Howell Ryan, Petersen S., Balzotti C., Rogers P., Jackson M., Hedrich A. Using WebGIS to Develop a Spatial Bibliography for Organizing, Mapping, and Disseminating Research Information: A Case Study of Quaking Aspen. *Rangelands*. 2019. Vol. 41. Iss. 6. P. 244–247. doi: 10.1016/j.rala.2019.10.001.
13. Nikita E., Mardini M., Mardini M., Tsimopoulou Ch., Karligkioti A. Bi(bli)oArch: An open-access bibliographic database for human bioarchaeological studies in the Eastern Mediterranean and Middle East. *Journal of Archaeological Science: Reports*. 2021. Vol. 39. doi: 10.1016/j.jasrep.2021.103151.
14. Batagelj V., Maltseva D. Temporal bibliographic networks. *Journal of Informetrics*. 2020. Vol. 14. Iss. 1. doi: 10.1016/j.joi.2020.101006.
15. Kastrin A., Hristovski D. Disentangling the evolution of MEDLINE bibliographic database: A complex network perspective. *Journal of Biomedical Informatics*. 2019. Vol. 89. P. 101–113. doi: 10.1016/j.jbi.2018.11.014.
16. Frandsen T., Gildberg F., Tingleff E. Searching for qualitative health research required several databases and alternative search strategies: a study of coverage in bibliographic databases. *Journal of Clinical Epidemiology*. 2019. Vol. 114. P. 118–124. doi: 10.1016/j.jclinepi.2019.06.013.
17. Cuschieri S., Grech V., Calleja N. WASP (Write a Scientific Paper): The use of bibliographic management software. *Early Human Development*. 2019. Vol. 128. P. 118–119. doi: 10.1016/j.earlhumdev.2018.09.012.
18. Stec E. Maintaining Nursing Knowledge Using Bibliographic Management Software. *Perioperative Nursing Clinics*. 2012. Vol. 7. Iss. 2. P. 195–200. doi: 10.1016/j.cpen.2012.02.004.
19. Nowiński A. Current Bibliography Research Information Systems in Poland. *Procedia Computer Science*. 2014. Vol. 33. P. 174–178. doi: 10.1016/j.procs.2014.06.029.
20. Curk L. Implementation of the Evaluation of Researchers' Bibliographies in Slovenia. *Procedia Computer Science*. 2019. Vol. 146. P. 72–83. doi: 10.1016/j.procs.2019.01.082.

References (transliterated)

1. Willer M., Dunsire G. *Bibliographic Information Organization in the Semantic Web*. Chandos Publishing, 2013, 318 p., doi: 10.1016/B978-1-84334-731-6.50001-6.
2. Tsybulnyk S., Bidnyk D. Designing the architecture of an automated bibliographic system. *Bulletin of the National Technical University „KhPI”. Series: New solutions in modern technology*. Kharkiv: NTU „KhPI”, 2021, no. 2 (8), pp. 83–89, doi: 10.20998/2413-4295.2021.02.12.
3. Campos-Asensio C. How to develop a bibliographic search strategy. *Enfermería Intensiva (English ed.)*, 2018, Vol. 29, Iss. 4, pp. 182–186, doi: 10.1016/j.enfie.2018.09.001.
4. Fernández-Ananín S., Rodríguez J., Soler E. Then, how do I document the idea? Bibliographic search engines. *Cirugía Española (English Edition)*, 2022, doi: 10.1016/j.cireng.2022.04.020.
5. Fayemi P.-E., Duci S., Fayolle T., Maranzana N., Bersano G. Ideality & Bio-Inspired Based Collaborative Bibliographic Search Method. *Procedia CIRP*, 2016, Vol. 39, pp. 138–143, doi: 10.1016/j.procir.2016.01.179.
6. Yun J. Generalization of bibliographic coupling and co-citation using the node split network. *Journal of Informetrics*, 2022, Vol. 16, Iss. 2, doi: 10.1016/j.joi.2022.101291.
7. Rico M., Vila-Suero D., Botezan I., Gómez-Pérez A. Evaluating the impact of semantic technologies on bibliographic systems: A user-centred and comparative approach. *Journal of Web Semantics*, 2019, Vol. 59, doi: 10.1016/j.websem.2019.03.001.
8. Lee O.-J., Jeon H.-J., Jung J. Learning multi-resolution representations of research patterns in bibliographic networks. *Journal of Informetrics*, 2021, Vol. 15, Issue 1, doi: 10.1016/j.joi.2020.101126.
9. Zhu Y., Yan E. Searching bibliographic data using graphs: A visual graph query interface. *Journal of Informetrics*, 2016, Vol. 10, Iss. 4, pp. 1092–1107, doi: 10.1016/j.joi.2016.09.005.
10. Zhu Y., Yan E., Song I.-Y. A natural language interface to a graph-based bibliographic information retrieval system. *Data & Knowledge Engineering*, 2017, Vol. 111, pp. 73–89, doi: 10.1016/j.datak.2017.06.006.
11. Sile L. Entanglement of bibliographic database content and data collection practices: Rethinking data integration using findings from a European study. *Procedia Computer Science*, 2019, Vol. 146, pp. 201–207, doi: 10.1016/j.procs.2019.01.094.
12. Howell Ryan, Petersen S., Balzotti C., Rogers P., Jackson M., Hedrich A. Using WebGIS to Develop a Spatial Bibliography for Organizing, Mapping, and Disseminating Research Information: A Case Study of Quaking Aspen. *Rangelands*, 2019, Vol. 41, Iss. 6, pp. 244–247, doi: 10.1016/j.rala.2019.10.001.
13. Nikita E., Mardini M., Mardini M., Tsimopoulou Ch., Karligkioti A. Bi(bli)oArch: An open-access bibliographic database for human bioarchaeological studies in the Eastern Mediterranean and Middle East. *Journal of Archaeological Science: Reports*, 2021, Vol. 39, doi: 10.1016/j.jasrep.2021.103151.

14. Batagelj V., Maltseva D. Temporal bibliographic networks. *Journal of Informetrics*, 2020, Vol. 14, Iss. 1, doi: 10.1016/j.joi.2020.101006.
15. Kastrin A., Hristovski D. Disentangling the evolution of MEDLINE bibliographic database: A complex network perspective. *Journal of Biomedical Informatics*, 2019, Vol. 89, pp. 101–113, doi: 10.1016/j.jbi.2018.11.014.
16. Frandsen T., Gildberg F., Tingleff E. Searching for qualitative health research required several databases and alternative search strategies: a study of coverage in bibliographic databases. *Journal of Clinical Epidemiology*, 2019, Vol. 114, pp. 118–124, doi: 10.1016/j.jclinepi.2019.06.013.
17. Cuschieri S., Grech V., Calleja N. WASP (Write a Scientific Paper): The use of bibliographic management software. *Early Human Development*, 2019, Vol. 128, pp. 118–119, doi: 10.1016/j.earlhumdev.2018.09.012.
18. Stec E. Maintaining Nursing Knowledge Using Bibliographic Management Software. *Perioperative Nursing Clinics*, 2012, Vol. 7, Iss. 2, pp. 195–200, doi: 10.1016/j.cpen.2012.02.004.
19. Nowiński A. Current Bibliography Research Information Systems in Poland. *Procedia Computer Science*, 2014, Vol. 33, pp. 174–178, doi: 10.1016/j.procs.2014.06.029.
20. Curk L. Implementation of the Evaluation of Researchers' Bibliographies in Slovenia. *Procedia Computer Science*, 2019, Vol. 146, pp. 72–83, doi: 10.1016/j.procs.2019.01.082.

Відомості про авторів (About authors)

Цибульник Сергій Олексійович – кандидат технічних наук, доцент, КПІ ім. Ігоря Сікорського, доцент кафедри комп'ютерно-інтегрованих оптичних та навігаційних систем; м. Київ, Україна; ORCID: 0000-0002-4462-0936; e-mail: tsybulnik.s.a@gmail.com.

Tsybulnyk Serhii – Candidate of Technical Sciences (Ph. D.), Docent, Associate Professor, Department of Computer-Integrated Optical and Navigation Systems, Igor Sikorsky Kyiv Polytechnic Institute, Kyiv, Ukraine; ORCID: 0000-0002-4462-0936; e-mail: tsybulnik.s.a@gmail.com.

Бідник Данило Сергійович – студент кафедри комп'ютерно-інтегрованих оптичних та навігаційних систем; м. Київ, Україна.

Bidnyk Danylo – student, Department of Computer-Integrated Optical and Navigation Systems, Igor Sikorsky Kyiv Polytechnic Institute, Kyiv, Ukraine.

Півторак Діана Олександрівна – кандидат технічних наук, доцент, КПІ ім. Ігоря Сікорського, доцент кафедри комп'ютерно-інтегрованих оптичних та навігаційних систем; м. Київ, Україна; ORCID: 0000-0003-3708-5610; e-mail: tsybulnik.s.a@gmail.com.

Pivtorak Diana – Candidate of Technical Sciences (Ph. D.), Docent, Associate Professor, Department of Computer-Integrated Optical and Navigation Systems, Igor Sikorsky Kyiv Polytechnic Institute, Kyiv, Ukraine; ORCID: 0000-0003-3708-5610; e-mail: tsybulnik.s.a@gmail.com.

Будь ласка, посилайтесь на цю статтю наступним чином:

Цибульник С. О., Бідник Д. С., Півторак Д. О. Розроблення автоматизованої бібліографічної системи. *Вісник Національного технічного університету «ХПІ»*. Серія: Нові рішення в сучасних технологіях. – Харків: НТУ «ХПІ». 2022. № 2 (12). С. 54–60. doi:10.20998/2413-4295.2022.02.08.

Please cite this article as:

Tsybulnyk S., Bidnyk D., Pivtorak D. Development of an automated bibliographic system. *Bulletin of the National Technical University "KhPI". Series: New solutions in modern technology*. – Kharkiv: NTU "KhPI", 2022, no. 2(12), pp. 54–60, doi:10.20998/2413-4295.2022.02.08.

Надійшла (received) 23.05.2022