Section: Information Technology, Cyber Security and Computer Engineering

DOI 10.70286/EOSS-24.11.2025.002

INFORMATION ARCHITECTURE AND UI/UX DESIGN OF A MOBILE APPLICATION FOR COWORKING SPACES

Ziuziun Vadym
PhD, Associate Professor
ORCID: 0000-0001-6566-8798
Demchuk Ganna
Master's degree student
Department of Technologies Management
Taras Shevchenko National University of Kyiv, Ukraine

Abstract. The study presents a comprehensive approach to designing a mobile application for booking coworking spaces, integrating information architecture, structured data modeling, and modern UI/UX design principles. The work examines the development of a three-tier architecture based on the Firebase infrastructure, which ensures scalability, serverless logic processing, and secure data storage in Firestore. It outlines the key aspects of creating conceptual and logical database models that define the entities, attributes, and relationships essential for the proper operation of the booking service. Significant attention is devoted to developing the user interface, designed according to the principles of minimalism, intuitive navigation, and adaptability, as well as employing Figma as the primary design tool. The study also highlights the integration of artificial intelligence tools used to generate personalized recommendations and analyze user feedback, thereby enhancing the quality and convenience of user interaction with the service. The proposed solution demonstrates the potential of combining cloud technologies, well-structured system architecture, and contemporary design approaches to create a competitive digital product.

Keywords: information system, information architecture, IT project, mobile application, database, Firebase, React Native, Figma, coworking.

Introduction. The rapid development of digital technologies and the growing prevalence of remote work have created a demand for effective tools that support the organization of workspaces beyond traditional office environments. Coworking spaces have become an important component of modern infrastructure; however, users increasingly expect fast, convenient, and personalized access to the services these locations provide. This drives the need for intelligent mobile solutions capable of automating booking processes, enabling interactive engagement, and delivering a high-

quality user experience. The relevance of developing the Coworkly mobile application lies in the necessity to design a comprehensive information architecture that integrates a scalable Firebase cloud infrastructure, flexible data models, and an intuitive UI/UX design. Another essential aspect is the incorporation of artificial intelligence technologies to generate personalized recommendations, analyze user feedback, and improve service quality. Ensuring data security, user-friendly navigation, and system stability has become a critical factor in maintaining competitiveness. Therefore, the project is highly relevant given the increasing demand for mobile booking services, the need to optimize user interactions, and the growing importance of efficient management of modern coworking spaces.

The aim of the study is to develop the information architecture, as well as the conceptual and logical database models, and to create an intuitive and functional UI/UX design for the Coworkly mobile application used to book coworking spaces. The implementation of this set of solutions is intended to ensure user convenience, high system reliability, and the potential for further expansion of the application's functionality.

Research results. The software for the coworking space booking platform is implemented using Google Firebase [1; 2], a modern cloud platform that provides a full stack for developing, scaling, and maintaining web and mobile applications. This approach enables rapid project deployment, reduces the cost of maintaining server infrastructure, and ensures high reliability, scalability, and security.

The system architecture consists of three logical layers:

- the client interface;
- cloud server functions;
- the database (Firebase Realtime Database [3]).

Client layer. This layer is implemented using the React Native framework [4; 5] for the mobile application.

The user interface includes the following features:

- creating an account and signing in via email or social media;
- selecting the interface language (Ukrainian or English);
- filtering and browsing available coworking spaces;
- booking workstations through an interactive calendar;
- online payments;
- leaving ratings and comments;
- viewing recommendations based on the user’s previous activity.

The frontend interacts directly with Firebase through official SDKs, which simplifies integration and reduces the number of intermediaries between the client and the database.

Cloud logic layer. Instead of a traditional server environment, this project uses Firebase Cloud Functions [6], which serve as a serverless mechanism for processing business logic:

1. Processing booking requests (checking availability, creating database records, and generating receipts).

2. Processing payments (integration with payment services such as Stripe or Fondy).

3. Authentication and security (validating access tokens and managing access rights).

4. Data change triggers (automatic updates of statistics, ratings, history, and other related information).

5. Invoking OpenAI AI interfaces (transmitting data to generate personalized recommendations).

Cloud Functions [6] scale automatically according to the system load, require no manual administration, and maintain a high level of security, as they operate within the protected environment of Google Cloud.

Database layer. Firebase Firestore [7] is used as the data storage solution—a document-oriented cloud database that supports real-time data synchronization between the client and the server.

Its advantages include:

1. High access speed.
2. Automatic replication and scalability.
3. Support for offline mode on client devices.
4. A flexible system of access security rules.

The database will store:

- user data (accounts, settings);
- information about coworking spaces (location, availability, description);
- bookings, history, and payments;
- reviews and ratings;
- saved recommendations for the use.

A key feature of this IT project is its integration with artificial intelligence, specifically through the OpenAI API. This integration is implemented via Cloud Functions [6], which:

- analyze booking history, usage patterns, and user preferences;
- generate personalized recommendation texts for new coworking spaces based on this analysis;
- additionally, process user reviews by classifying them (positive, negative, neutral) and performing thematic analysis.

This enables intelligent functionality that goes beyond simply recording data, providing users with valuable insights – for example, recommending less crowded spaces or those with high ratings at specific times.

Data security and handling include:

1. Authentication is implemented via Firebase Authentication [8; 9], supporting email/password as well as Google and Facebook logins.

2. Data access is controlled through Firebase Security Rules [10].

3. All data is transmitted over encrypted channels (HTTPS).

4. External certified platforms that comply with PCI DSS standards are used to store payment information.

The importance of establishing an effective information architecture for an IT project [11], particularly in developing an online service, is determined by the rapid pace of technological advancement and the increasing demands of users for simplicity and efficiency. In light of the constant changes in the online environment, optimal organization of information becomes a crucial component for successful project implementation. Ensuring easy access and an intuitive interface is essential for attracting and retaining users. The information architecture reflects the simplification and optimization of interactions between users and the service, which is critical in a competitive digital landscape [12].

Fig. 1 presents a graphical model of the information system architecture.

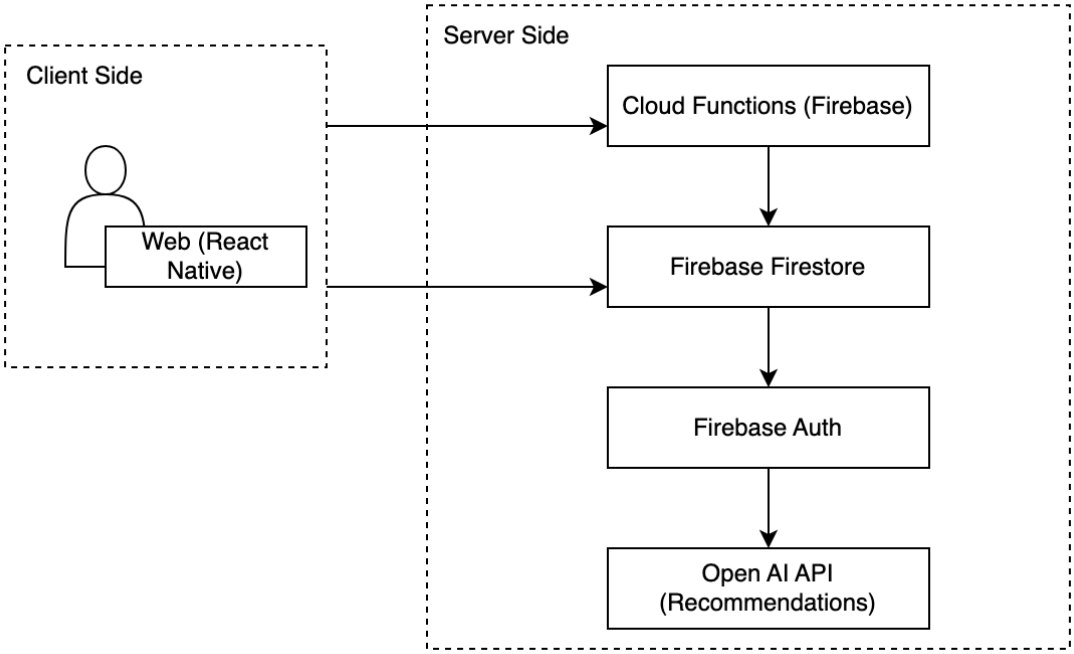

Figure 1. Graphical model of the information system architecture

As part of the research on developing the mobile application Coworkly, a conceptual database model was created [13] (Fig. 2), serving as the foundation for formalizing the system's structure. The diagram illustrates the key entities involved in the service's functionality. These include the user, coworking space, workspace, booking, review, and favorite. Each of these entities has its own set of attributes describing its characteristics: for example, a user has a first name, last name, email, phone number, and password, while a coworking space includes its name, address, capacity, description, and operating hours.

The relationships between objects clearly define the logic of interaction: a single user can leave multiple reviews, but each review is associated with only one user; each workspace belongs to a specific coworking space, and a booking links a user, a

coworking space, and a workspace within a defined time period. The «favorite» entity allows users to save their preferred coworking spaces.

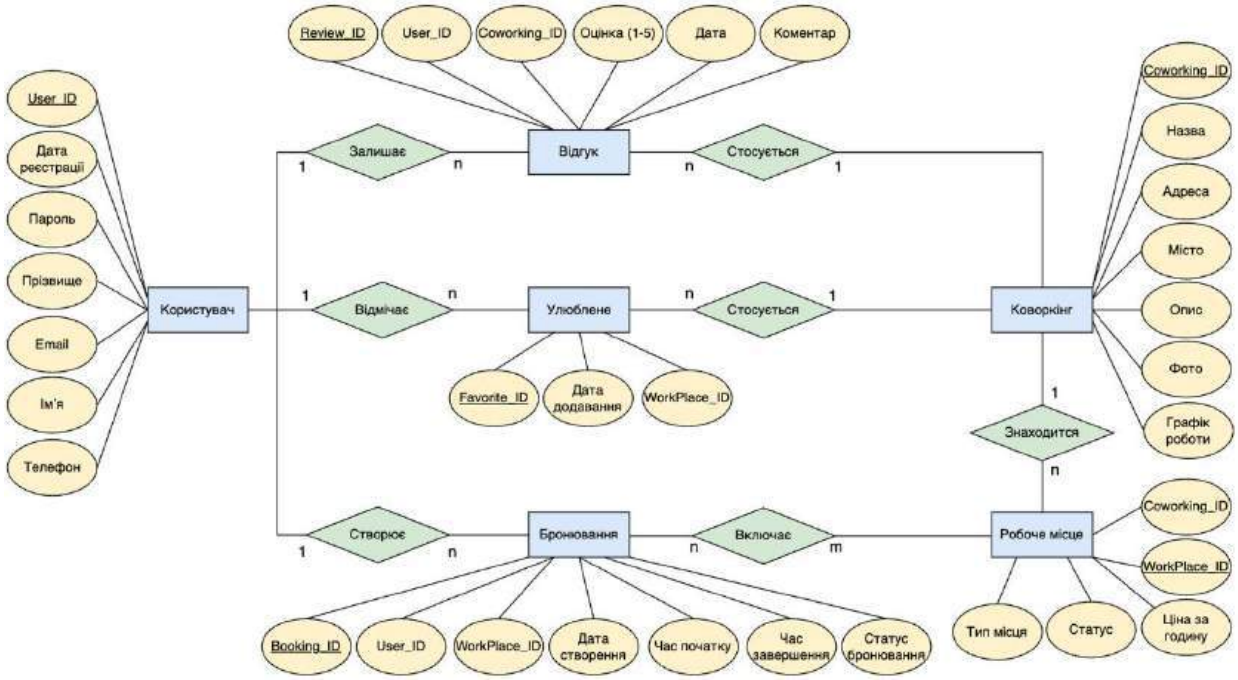

Figure 2. Conceptual database model

The logical database model [14] (Fig. 3) represents the next step after creating the conceptual model of the information system. It provides a more formalized and detailed data structure, closer to implementation within a database management system, but it does not yet include specific technical solutions for physical storage.

Figure 3. Logical database model

Its primary goal is to describe the logic of data organization, table structures, attributes (fields), data types, as well as the keys and constraints that ensure data integrity and consistency. The logical database model illustrates how the data from the conceptual model will be organized into tables, what relationships exist between them (one-to-one, one-to-many, many-to-many), and which fields serve as primary or foreign keys. This allows for a deeper understanding of the technical aspects of building an information system, anticipating potential issues with query execution, optimizing the database structure, and ensuring scalability.

The importance of the logical model lies primarily in its role as the technical foundation for creating a real database. It enables developers to design a convenient and efficient structure that ensures fast access to information, data integrity, protection, and consistency during updates. Additionally, this model is critical for writing queries, creating APIs, designing data interaction interfaces, and integrating with other systems.

Based on the logical model, one can determine which tables will be created in the database, the attributes they will have, which fields are used to establish relationships between tables, how one-to-many or many-to-many relationships are implemented, and how data uniqueness is enforced. For example, in a coworking system, this allows for accurately tracking who booked which workspace, in which coworking space, for what time, and the status of the booking.

The development of both conceptual and logical database models is a critical stage in building a web platform for automated checking and grading of written tests. This approach allows all necessary information to be stored in a structured manner for efficient system operation. Proper data conceptualization ensures convenient access and quick retrieval of required information. A well-designed logical database guarantees data correctness and consistency, which is essential for the accuracy and reliability of the grading process. It also facilitates future system scaling and modifications.

In the process of creating the mobile application interface for the Coworkly coworking network, modern UX/UI design principles [15] were followed, focusing primarily on usability and clarity for the end user. Although the development team does not have formal design education, the goal was to create an interface that provides intuitive navigation, ease of interaction, and a positive user experience.

The design concept was based on the principle of an intuitive interface, ensuring a logical arrangement of all interface elements. Key functions – such as searching for coworking spaces, booking workspaces, viewing profiles, and making online payments – were positioned so that users could easily find and use them without additional instructions.

Additionally, principles of minimalism and responsiveness were applied, which are highly relevant in modern mobile design. The interface features simple color schemes, clear typography, and adequate spacing between elements, enhancing readability and improving user interaction with the app.

For prototype development, the cloud-based tool Figma [16; 17] was used, one of the most popular platforms for interface design. Its advantages include ease of use, an intuitive interface, versatility, and the ability to work directly in the browser. Another key benefit of Figma [18] is its support for collaborative work, allowing all edits and updates to be synchronized in real time. This is especially valuable in team-based development or when receiving feedback from a manager or team members.

The design approach was user-centered, meaning the interface was created with the needs, expectations, and behaviors of potential users in mind. Interaction scenarios were carefully planned – from registration to completing a booking. Each step was organized logically, and processes were simplified to avoid information overload or unnecessary actions.

Special attention was given to the visual structure. Content was grouped into clear, understandable blocks, and key actions were highlighted with color accents – for example, booking buttons were emphasized with bright colors.

As a result, an interface was created that meets modern standards, prioritizes user convenience, and is ready for implementation as a fully functional software product.

Below are screenshots of the app with the developed interfaces (Figs. 4–6).

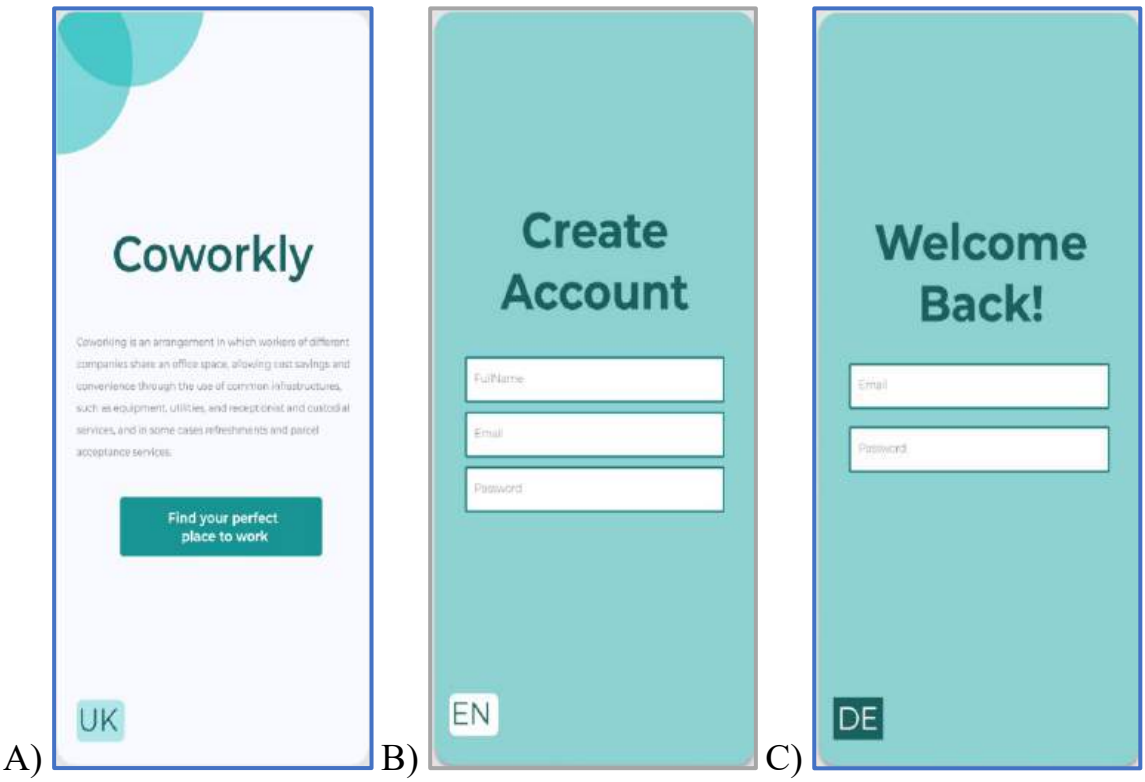

Figure 4. A) screen after the app loads; B–C) user registration and authentication screens

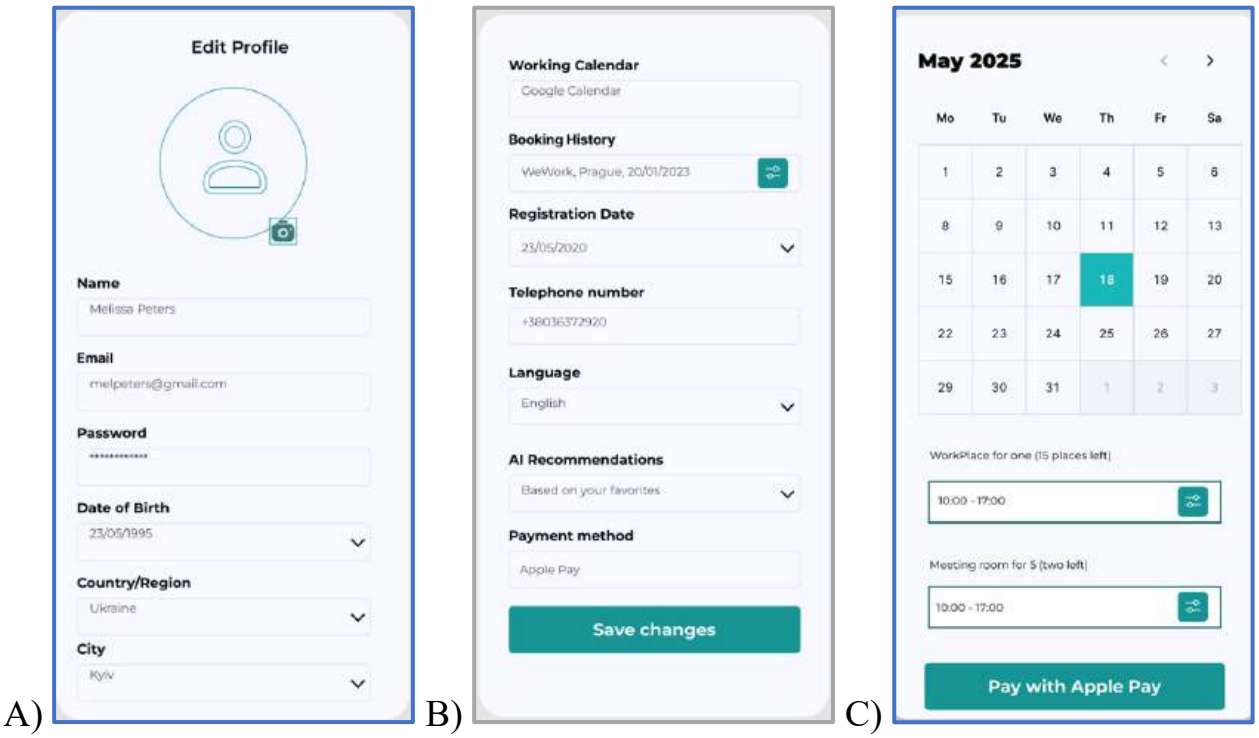

Figure 5. A–B) user profile editing screens; C) booking date screen

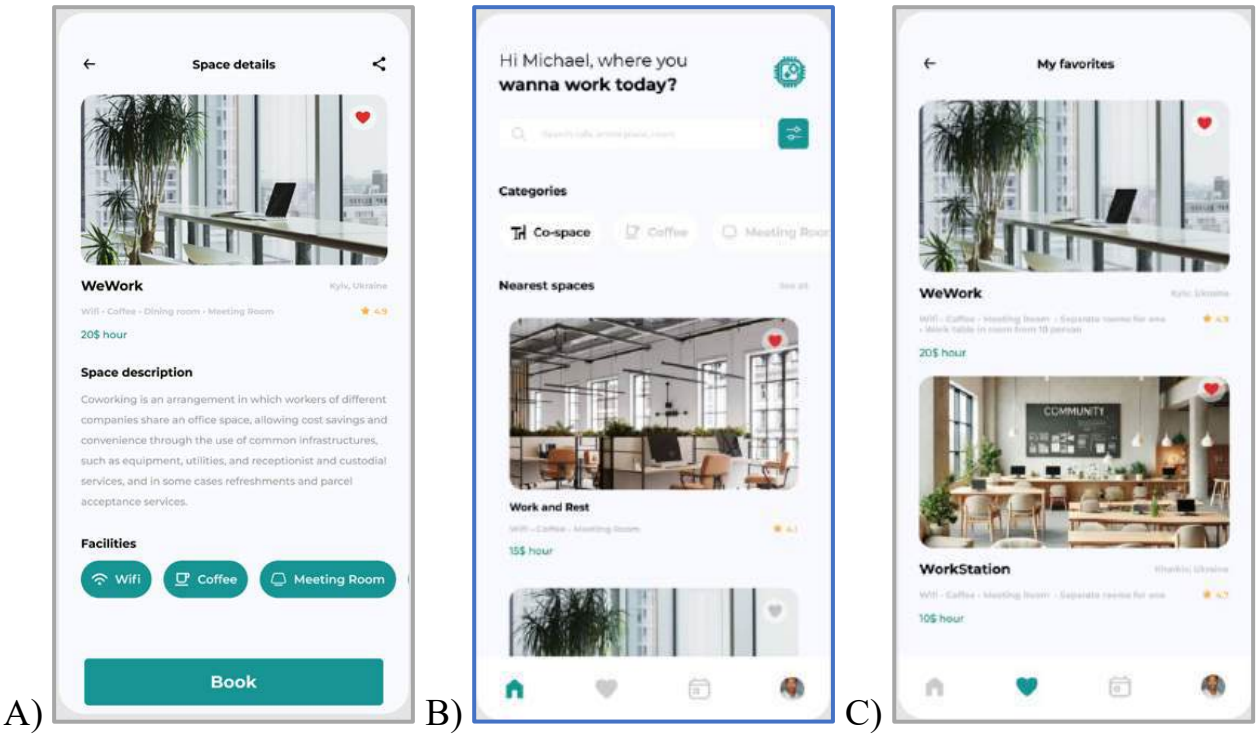

Figure 6. A) screen showing a detailed description of the coworking space selected by the user; B) screen with a list and filter options for available nearby coworking spaces; C) «Favorites» screen, displaying coworking spaces liked and saved by the user

Conclusion. During the course of this research, a comprehensive information architecture for the Coworkly mobile application was developed, ensuring efficient interaction between users, cloud services, and the database. Implementing the program logic using Firebase Cloud Functions enabled the creation of a scalable serverless

solution that automatically handles bookings, payments, and user requests. The conceptual and logical database models clearly defined the structure of entities and relationships necessary for correct data storage and processing, ensuring data integrity, fast access, and the potential for future system expansion.

Significant attention was given to creating an intuitive and responsive UI/UX design, focused on usability and clear navigation. The integration of artificial intelligence tools enhances the application's functionality by providing personalized recommendations and analyzing user feedback. Leveraging modern digital technologies ensures high reliability and system security.

The developed application meets current digital market requirements and demonstrates potential for scalability. The project underscores the importance of combining technical solutions, design approaches, and intelligent algorithms in modern mobile services. Overall, the work establishes a solid foundation for the further development and enhancement of the Coworkly platform.

References

1. Cloud Functions for Firebase. (2023). URL: https://developers.google.com/assistant/console/firebase-services (access date: 02.11.2025)
2. GOOGLE. Firebase. Documentação do FirebaseRealtimeDatabase. (2024). URL: https://firebase.google.com/docs/database. (access date: 02.11.2025)
3. Moroney, L. (2017). The Firebase Realtime Database. In: The Definitive Guide to Firebase. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-2943-9_3
4. What Is React Native? Beginner's Guide + FAQ. (2025). URL: https://www.coursera.org/articles/what-is-react-native (access date: 08.11.2025)
5. React Native Firebase. (2025). URL: https://rnfirebase.io (access date: 08.11.2025)
6. Cloud Functions for Firebase. (2025). URL: https://firebase.google.com/docs/functions (access date: 10.11.2025)
7. Bernardino, S.; De Jesus Rodrigues De Barros, L. (2025). Firebase and its usage in android and ios applications. Revista Interface Tecnológica, Taquaritinga, SP, 21(1), 279–287, 2025. https://doi.org/10.31510/infa.v21i1.1978.
8. ReleaseNotes Firebase. (2025). URL: https://firebase.google.com/support/releases (access date: 12.11.2025)
9. Firebase Studio. (2025). URL: https://firebase.google.com/docs/studio (access date: 12.11.2025)
10. Ralenko, V, & Stoiev, Ye. (2025). Using Firebase Studio in developing React applications. Scientific and Practical Journal «Materials of Scientific Conferences of the Petro Mohyla Black Sea National University». URL: https://doi.org/10.34132/mspc2025.01.14.14 (access date: 12.11.2025)
11. Kolomiiets, A., Miroshnychenko, I., Ziuziun, V., Datsenko, N., & Kmytiuk, T. (2024). Development of Project Management Models for Information Systems to Improve Website SEO Metrics. XI International Scientific Conference "Information Technology and Implementation" (IT&I 2024). CEUR Workshop Proceedings, Vol-3909, 334-345. URL: https://ceur-ws.org/Vol-3909/Paper_27.pdf (access date: 12.11.2025)

12. Ziuziun, V., & Petrenko, N. (2025). AI-Enhanced System Design for Agile Sprint Management and Velocity Prediction. 2025 IEEE 5th International Conference on Smart Information Systems and Technologies (SIST), Astana, Kazakhstan, 2025, pp. 1-6. https://doi.org/10.1109/SIST61657.2025.11139157
13. Ziuziun, V., & Bredikhin, D. (2025). Rozrobka informacijnoi' struktury projektu stvorennja veb-platformy dlja pidvyshhennja ekologichnoi' kul'tury naselennja [Development of the information structure of the project for creating a web platform to improve the ecological culture of the population]. Collection of Scientific Papers of Admiral Makarov National University of Shipbuilding, 1(499), 126-136. https://doi.org/10.15589/znp2025.1(499).18
14. Ziuziun, V., & Petrenko, N. (2024). Formation of conceptual and logical databases for the project of creating a web platform for automated verification and evaluation of written tests. XXI International scientific and practical conference «Project management in the development of society», Kyiv, KNUBA, 32-37. URL: https://library.knuba.edu.ua/books/zbirniki/Upravlinnya_proektami_u_rozvitku_suspilstvaa_XXI.pdf#page=32 (access date: 08.11.2025)
15. Rahmadani, F., & Susetyo, Y. A. (2025). The UI/UX Design of a Mobile-Based Catering Application Using the Activity Centered Design Method. International Journal Software Engineering and Computer Science (IJSECS), 5(1), 29-39. https://doi.org/10.35870/ijsecs.v5i1.3396
16. Figma: The Collaborative Interface Design Tool. (2025). URL: https://www.figma.com (access date: 10.11.2025)
17. Green, T., & Brandon, K. (2024). Microinteractions in Figma. UX Design with Figma. Design Thinking. Apress, Berkeley, CA. https://doi.org/10.1007/979-8-8688-0324-6_8
18. Ziuziun, V., & Koziuk, Yu. (2025). Conceptual Modeling of an Information System for Professional Development and Promotion of Educational Services in the Field of Digital Design, 4(153), 156-163. https://doi.org/10.32782/1995-0519.2025.4.19

ДІЯЛЬНІСТЬ БІБЛІОТЕЧНИХ УСТАНОВ У НАПРЯМІ КРЕАТИВ-РЕКЛАМУВАННЯ ІНФОРМАЦІЙНИХ ПОСЛУГ

Федотова Оксана
д. іст. наук, ст. наук співробітник
Кафедра інформаційної діяльності
Маріупольський державний університет, Україна

Сучасне інформаційне суспільство характеризується стрімкими темпами розвитку новітніх технологій та поширення даних [7]. В умовах зростання інформаційних потоків традиційні підходи до рекламування інформаційної продукції та послуг втрачають ефективність. Бібліотеки стикаються з проблемою не просто інформувати про свої ресурси, але й зробити це в такий спосіб, аби в