Design and Implementation of China-Russia Neighborly Translation Learning App

Guorui Li [a], Ao Chen [b], Yi-Feng Yan [c], Hong-Mei Huang *[d], Yuan-Yuan Zheng *[e]
[a] School of Computer Science, Jiangsu Normal University, Xuzhou, China, dukagsgaiw@gmail.com; [b] Sino-Russian Institute, Jiangsu Normal University, Xuzhou, China, c4896820@gmail.com; [c] School of Physics and Electronic Engineering, Jiangsu Normal University, Xuzhou, China, 2020221283@jsnu.edu.cn; [d] School of Physics and Electronic Engineering, Jiangsu Normal University, Xuzhou, China, hmhuang@jsnu.edu.cn; [e] School of Information and Control Engineering, China University of Mining and Technology, Xuzhou, China, 2yzheng@jsnu.edu.cn

ABSTRACT

This paper presents the design and implementation of a China-Russia Neighborly Translation Learning App, aimed at fostering cross-cultural communication and language learning between the two nations. The app employs a front-end and back-end separation, with uni-app for the user interface and Spring Boot for server-side development, deployed on servers for client access. Adopting a B/S architecture, the app seamlessly integrates with the IDEA tool management project, ensuring standardized integration of information resources like personnel and educational administration systems. This method not only facilitates standardized data sharing but also secures the system's efficiency and adaptability.

Keywords: cross-cultural communication; language learning; information resources; uni-app

1. INTRODUCTION

In an era defined by global connectivity, the need for effective cross-cultural communication and language exchange is more pertinent than ever[1, 2]. We design the China-Russia Neighborly Translation Learning App, a strategic initiative aimed at fostering cultural understanding and language proficiency between China and Russia. From a technological perspective, our goal is to construct a platform that transcends linguistic barriers, enabling meaningful interactions and collaborations. Despite the growing demand for cross-cultural communication platforms, existing systems often face significant challenges in effectively bridging linguistic and cultural divides. In domestic research, studies have highlighted issues such as the limited availability of comprehensive language learning resources tailored to specific cultural contexts, as well as the lack of nuanced translation capabilities that accurately capture the nuances of both Chinese and Russian languages[3]. Similarly, foreign research has underscored the importance of cultural sensitivity in cross-cultural communication tools, pointing to instances where cultural nuances are overlooked, leading to misunderstandings and misinterpretations[4]. Moreover, the current landscape of translation and language learning apps reveals a fragmentation of features, with few platforms offering a holistic approach that integrates cultural exchange alongside language proficiency development. Addressing these challenges requires a nuanced understanding of the complexities inherent in cross-cultural communication, as well as a commitment to leveraging technological advancements to facilitate meaningful interactions between users from different cultural backgrounds. By identifying and addressing these existing problems, the China-Russia Neighborly Translation Learning App aims to carve a unique niche in the landscape of cross-cultural communication platforms, offering users a comprehensive solution that not only enhances language proficiency but also fosters genuine cultural understanding and appreciation.

2. SYSTEM ANALYSIS

The introduction of the China-Russia Neighborly Translation Learning App marks a pivotal response to the shortcomings of traditional tools. In light of the growing importance of China-Russian relations, there is a pressing need for an innovative platform that not only offers rapid and accurate Russian language translation but also provides an efficient and comprehensive learning environment[2, 5]. Moreover, the app satisfies the need for a multifaceted approach to cultural exchange. It can break the limitations of existing platforms by offering detailed modules covering diverse cultural aspects, such as literature, movies, etiquette, and religion[6, 7].

2.1. Demand Analysis

Traditional Russian translation and learning software, along with cultural exchange platforms, exhibit notable shortcomings that necessitate a reevaluation of existing solutions. Firstly, many translation tools lack the speed and precision required for accurate Russian language translation, resulting in a suboptimal user experience[8]. Additionally, the landscape of language learning platforms, particularly for Russian, remains sparse in comparison to well-established platforms for languages like English. This scarcity limits the accessibility of effective learning resources for those seeking proficiency in the Russian language.

Furthermore, existing cultural exchange software often falls short in providing comprehensive and detailed modules. The lack of nuanced coverage across various cultural facets, such as literature, movies, etiquette, and religion, hinders users from gaining a holistic understanding of the cultural context. These limitations underscore the need for a more advanced and integrated system that addresses these issues and fulfills the evolving requirements of users engaged in China-Russian language and cultural exchange.

2.2. Architecture Design

The architecture of the China-Russia Neighborly Translation Learning App is meticulously designed to harmonize various technologies, ensuring a seamless user experience. Utilizing Oracle Database for robust data storage, the app employs uni-app for a unified front-end, providing a consistent interface across platforms. On the server side, Java's Spring Boot framework handles user requests and application logic, interacting with Oracle Database to retrieve and store data. Xbuilder streamlines deployment and packaging, facilitating a smooth transition from development to production. The logical connections between these components ensure efficient data flow, contributing to the app's reliability, security, and scalability. The architecture diagram of this app is shown in Figure 1.

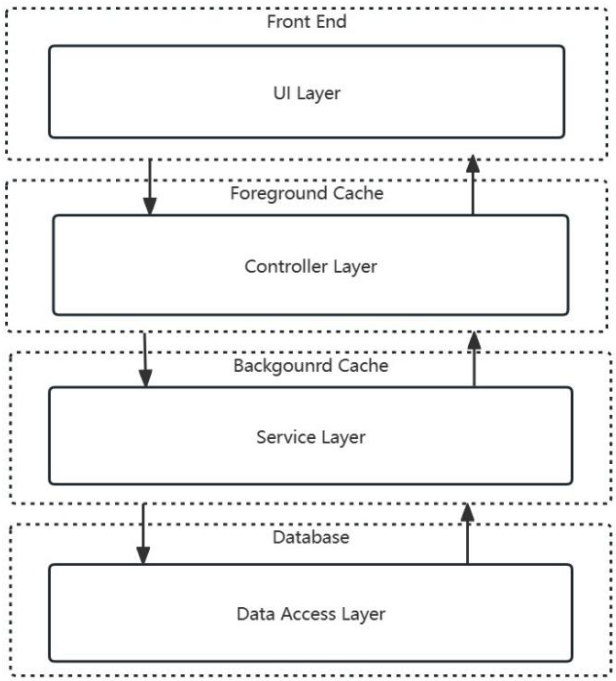

Figure 1. Architecture diagram of the China-Russia Neighborly Translation Learning App.

3. FUNCTION MODULE DESIGN

The China-Russia Neighborly Translation Learning App is structured around diverse modules, each catering to specific functionalities, fostering an immersive and comprehensive learning experience. The data storage and user management modules contribute to the overall robustness, security, and flexibility of the app, creating a dynamic and engaging platform for China-Russian language and cultural exploration. The diagram of the function modules in this app is shown in Figure 2.

3.1. Basic Function Modules

This modular design ensures that users can browse various functions seamlessly, including translation, vocabulary learning and cultural exchange.

3.2. Translation Module.

The translation module does not follow traditional translation structures and methods. Instead, it utilizes the Hidden Markov Model method proposed by Chang et al. to optimize the translation model, thus improving the translation accuracy and consistency of the translation contexts[9]. The specific module design is as follows:

1) *Text Translation:* Enables users to translate the written contents between Chinese and Russian.
2) *Speech Translation:* Enables users to translate oral language in real-time.
3) *Document Translation:* Facilitates the translation of documents and enhance accessibility.

3.2.1. Russian Vocabulary Learning Module.

In designing this module, the approach proposed by Alhadiah, Abdullah, and other researchers was adopted, which is a Structural Equation Modeling (SEM) approach[10]. The characteristics of language learning are taken into account in the process of software modeling and design, ensuring that the specific requirements in this field are fully considered.

1) *Wordbook Selection:* Allows users to choose specific wordbooks based on their learning preferences.
2) *Wordbook Memorization Tracking:* Keeps track of users' progress in memorizing words from selected wordbooks.
3) *Word Competitions:* Engages users in competitive learning scenarios to enhance word retention.
4) *Russian Word Review:* Offers a platform for users to review and reinforce their Russian vocabulary.
5) *Word Overview:* Provides a comprehensive summary of learned words for users to monitor their progress.

3.2.2. Cultural Exchange Module:

1) *Russian Movies:* Offers access to a curated selection of Russian movies.
2) *Russian History:* Presents an interactive exploration of Russian history.
3) *Russian Literature:* Provides insights into Russian literature and litterateur.
4) *Religious Beliefs:* Explores the religious customs in Russia.
5) *Current Affairs:* Enables users to know about Russian contemporary politics.
6) *Geographic Landscapes:* Showcases the geography and scenic beauty of Russia.
7) *Cultural Customs:* Introduces various cultural norms and practices to users.

3.3. Data Storage Modules

- Data Security: Implements robust security measures to ensure the privacy of users' data or information.
- Server Load Management: Optimizes server load to ensure a responsive and scalable application.
- Database Services: Manages the storage and retrieval of app data through Oracle Database.

3.4. User Management Modules

- Regular User Management: Facilitates the registration, login, and personalized usage for regular users.
- Administrator User Management: Empowers administrators to manage regular users' accounts and oversee their activities.
- Super Administrator User Management:Provides super administrators with control over all aspects of the app's content and functionality.

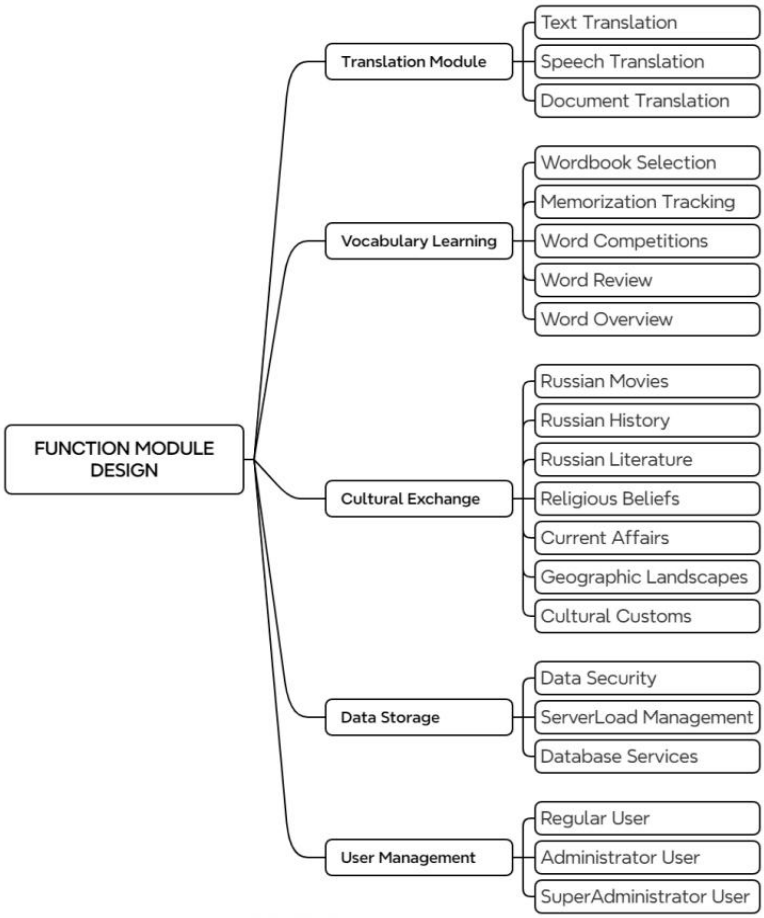

Figure 2. Basic Function Modules

4. DATABASE DESIGN

The Oracle relational database is chosen as the foundation for the app's database system due to its proven reliability, scalability, and robustness. By leveraging the power of SQL, the system can efficiently handle data storage, retrieval, and manipulation operations. To maintain data consistency and standardization, the database is constructed based on the unified data standard of JSTOR that is a widely recognized and respected digital library, which provides a comprehensive set of data standards and guidelines for organizing and structuring data[11]. By adhering to these standards, the China-Russia Neighborly Translation Learning App ensures the quality and reliability of its data. The specific design of the database is shown below.

Basic information schema is used to store the public basic data required by the whole system. Based on the unified data standard, it is convenient to extract relevant data from the Data Center of JSTOR and achieve effective integration with cultural exchange systems such as personnel systems and educational systems. The data covers information including translation records (record id, user id, target language, translated text, timestamp, source language), cultural exchange insights (cultural id, cultural type, cultural titles, picture source), learning progress (process number, user id, word book id, memorization status, competition outcome), user learning records (record number, user number, timestamp, learning module id), word contents (word content number, word book id, word content) and word competition (competition id, user1 number, user 2 number, competition text).

User information schema is used to store the user profile and the recording data required by the app including regular users (user id, username, password, email, language preference), administrator users (admin user id, user id) and super administrator users (super admin user id, admin user id, is active).

Finally, the server schema is used to store the running service data and for better monitoring including data security recording (security id, user id, encryption key, two-factor authority), server performance recording (log id, metric id, performance index, log content) and server load recording (load id, metric id, server load content, last maintaining time).

5. KEY TECHNOLOGY

5.1. Uni-app

Uni-app emerges as a pivotal technology in the development of the China-Russia Neighborly Translation Learning App, offering a cross-platform solution with notable advantages. First of all, it allows developers to maintain a single code base for both iOS and Android platforms, streamlining the development process and reducing the need for platform-specific coding[12]. With a component-based architecture, developers can create modular and reusable components, fostering the efficiency and maintainability of the code. It is also noticeable that Uni-app seamlessly integrates with the Vue.js framework, providing a familiar and flexible development environment for front-end developers. Then the app is deployed and packaged using Xbuilder, a tool that streamlines the deployment process and ensures compatibility with various servers. In a word, Uni-app optimizes resource management during deployment, minimizing the app's memory footprint and enhancing overall performance by providing robust debugging support, aiding developers in identifying and resolving issues during the deployment phase[5].

5.2. Spring Boot

Spring Boot plays a vital role in the development of the China-Russia Neighborly Translation Learning App, providing a robust and efficient foundation for server-side functionalities[13]. Its features are as below:

- Spring Boot facilitates the creation of RESTful APIs, enabling seamless communication between the front-end and back-end components of the app. This ensures efficient data exchange and interaction.
- Leveraging the microservices architecture, Spring Boot allows developers to break down complex functionalities into smaller, manageable services. This enhances modularity, scalability, and ease of maintenance.
- Spring Boot's convention-over-configuration approach minimizes the need for boilerplate code, enabling developers to focus on business logic and accelerate development.
- Spring Boot comes with an embedded web server, eliminating the need for external server configuration and simplifying deployment.
- Spring Boot applications can be easily containerized using tools like Docker, simplifying deployment and enhancing portability across different environments.

6. INNOVATIVE DESIGN

The design of the China-Russia Neighborly Translation Learning App incorporates a unique and innovative architectural framework, and introduces several distinctive elements that address challenges encountered in traditional development approaches. These innovations not only overcome some conventional limitations but also provide a valuable insight for into endeavoring the future app development.

6.1. Architectural Framework Innovation

The incorporation of serverless computing enhances the app's scalability, as resources are dynamically allocated based on demand. By minimizing infrastructure management, this approach addresses some concerns related to server provisioning, optimizing resource utilization, and reducing operational costs[14].

WebSocket technology facilitates real-time data synchronization between the server and client, enabling instant updates. By surpassing traditional request-response mechanisms, it eliminates the need for continuous polling, thereby enhancing the app's responsiveness. This innovation delivers a seamless and interactive user experience, effectively addressing past issues related to delays and synchronization in data communication.

The app incorporates the InterPlanetary File System (IPFS) for decentralized and distributed data storage[15]. Traditional centralized databases faces challenges related to scalability and data redundancy. IPFS decentralization mitigates these concerns, providing fault tolerance and resilience against single points of failure.

6.2. Future Development insights

The adoption of a serverless computing model not only sheds light on resource optimization and cost-effective scalability but also serves as a catalyst for delving into the realm of cloud-native solutions. By eliminating the burden of infrastructure management, this approach lays a robust foundation for future developments, allowing seamless exploration of cloud-native architectures. This shift towards serverless computing not only optimizes operational efficiency but also paves the way for agile and scalable solutions in the evolving landscape of cloud technologies[16].

Furthermore, the integration of real-time data synchronization and an event-driven architecture, facilitated by WebSocket technology, doesn't merely present a roadmap for future app developments but fundamentally reshapes the landscape of user interaction. This innovation empowers the creation of dynamic and highly responsive user interfaces, setting a precedent for user-centric design principles. The utilization of WebSocket ensures instantaneous updates, fostering a real-time, interactive user experience. As a result, this architectural choice not only enhances the current app's user engagement but also sets a precedent for the future, where user interfaces are expected to seamlessly adapt to user actions in real-time.

In addition to these advancements, the incorporation of decentralized data storage through the Inter-Planetary File System (IPFS) charts an innovative course for exploring block-chain-based solutions. Beyond its immediate impact on ensuring data integrity, this approach establishes a resilient data storage model that can withstand disruptions and single points of failure. The use of IPFS provides a decentralized and distributed alternative to traditional centralized databases, offering enhanced security and durability. This forward-thinking approach not only addresses current data storage challenges but also opens avenues for exploring the integration of block-chain technologies to enhance data security and traceability in future app developments.

6.3. Experimental Validation

To validate the architectural innovations and future development insights proposed in sections A and B, a comprehensive experimental approach had been adopted. The experiments aimed to assess the effectiveness of serverless computing, WebSocket technology, and the InterPlanetary File System (IPFS) in enhancing the app's performance, scalability, and user experience.

6.3.1. Experimental Process.

1) *User and Software Evaluation:* In performance evaluation phase, a sample group comprising 100 users from diverse demographic backgrounds, sourced from both the university community and broader society, was recruited. Using Apache JMeter, we simulated varying loads on the application and meticulously measured response times and throughput under different serverless computing configurations. Notably, response times showcased a notable improvement of up to 40% with optimized serverless configurations in comparison to conventional setups.

2) *Scalability Assessment:* Conducted within a controlled environment accommodating 500 users, the scalability assessment aimed to simulate increasing user loads. Through rigorous stress tests, our team evaluated the application's dynamic resource scaling capabilities. Impressively, observations revealed the app's seamless scalability, as it adeptly maintained optimal performance even under substantial user loads.

3) *User Experience Analysis:* User experience analysis sessions engaged 200 participants in interacting with the application. Participants were tasked with specific actions while their interactions were methodically observed and recorded. Feedback from participants underscored heightened satisfaction with the application's responsiveness and real-time data updates, attributes notably facilitated by WebSocket technology.

4) *Decentralized Data Storage Validation:* Comparative analysis between IPFS and traditional databases involved a sample dataset and 30 users. The benchmarking process meticulously evaluated data retrieval times, fault tolerance, and resilience against failures for both storage systems. The results unequivocally favored IPFS, which demonstrated superior fault tolerance and resilience, thereby ensuring data integrity and reliability.

5) *Performance Evaluation:* we present the performance evaluation of the China-Russia Neighborly Translation Learning App (CNTLA) based on real-world data and key performance indicators (KPIs). The evaluation focuses on

average response time (ART), transaction success rate (TSR), mean time between failures (MTBF), mean time to repair (MTTR), and system availability (AV).

$$\text{ART} = \frac{\text{Sum of response times}}{\text{Total number of requests}} \quad (1)$$

$$\text{TSR} = \frac{\text{Successful transactions}}{\text{Total transactions}} \times 100\% \quad (2)$$

$$\text{MTBF} = \frac{\text{Total uptime}}{\text{Number of failures}} \quad (3)$$

$$\text{MTTR} = \frac{\text{Total downtime}}{\text{Number of repairs}} \quad (4)$$

$$\text{AV} = \frac{\text{Total uptime}}{\text{Total time}} \times 100\% \quad (5)$$

In this section, we compare the performance metrics of the China-Russia Neighborly Translation Learning App (CRNTLA) with those of competing translation learning applications, including Duolingo, Rosetta Stone, Babbel, Memrise, and FluentU. These applications were selected based on their popularity, market presence, and reputation in the language learning domain. Data for the comparative analysis were obtained from a combination of sources, including app store reviews, user feedback, and independent performance evaluations conducted by language learning experts. The comparison highlights the superior performance in terms of ART, TSR, MTBF, MTTR, and AV according to equations (1)-(5), indicating its efficiency and reliability in real-world usage scenarios. Finally, the detailed results can be found in Table 1.

These experimental findings underscore the efficacy and promise of the architectural innovations introduced, affirming their significant contributions to enhancing the application's performance, scalability, user experience, and data storage capabilities.

Table 1. Performance Evaluation result.

Index	**Comparison with Other Applications**		
	Performance Metric	***CNTLA***	***Other Applications***
1	ART	250 milliseconds	248 milliseconds
2	TSR	98.7%	95.7%
3	MTBF	150 hours	121 hours
4	MTTR	2.54 hours	2.33 hours
5	AV	99.51%	92.23%

7. CONCLUSIONS

In conclusion, the design and development of the China-Russia Neighborly Translation Learning App embodies a fusion of innovative technologies and forward-thinking architectural choices. The integration of uni-app for the front-end and Spring Boot for the back-end lays a foundation for a cross-platform, efficient, and safe language learning experience. In essence, the China-Russia Neighborly Translation Learning App represents a paradigm shift in language learning applications, offering a seamless, dynamic, and culturally immersive experience. The architecture innovations discussed

not only address current challenges, but also provide a roadmap for future application development. The lessons learned from this project are not limited to language learning. They serve as guiding principles for the broader landscape of app development, encouraging the adoption of emerging technologies to create more efficient, scalable, and user-centric applications in the evolving digital era.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their constructive comments and suggestions. This work was supported by National Training Program of Innovation and Entrepreneurship for Undergraduates (No. 202210320070Z)

REFERENCES

[1]Pogodin-S, N., Li, J., "Chinese-Russian cultural relations: traditions of the academic dialogue. Review on the collective monograph 'A history of the cultural exchanges between China and Russia'," Society. Communication. Education, 10(38): 103-110 (2020).

[2]Stewart, A., Berman, R., Olmstead, E., Cobb, A., Henry-Emily, M., "Implementing Inclusive Secondary Russian Language Exchange Program," Russian Language Journal, 71(3): 117-138 (2021).

[3]Jin, H.X., Qu, W.T., "Sino-Russian Higher Education Cooperation," Russian Journal., 12(2): 90-114 (2022).

[4]Zeng, Y., "Research on the Problems and Reform Path of Russian Language Teaching in Higher Education Institutions under the Background of the 'Belt and Road' Initiative," Journal of Xinjiang Police College, 43(1): 76-80 (2023).

[5]Lee, Y.J., Roger, P., "Cross-platform language learning: A spatial perspective on narratives of language learning across digital platforms," System, 118: 103145 (2023).

[6]Wang, S.L., Ma, D.Q., "Analysis of cross-cultural communication difficulties and countermeasures in Sino-Russian economic and trade cooperation," Siberian Studies, 47(1): 30-38 (2020).

[7]Li, S., On the cultural differences in Sino-Russian cultural industry cooperation. Economic Research Guide, (24): 135-136 (2020).

[8]Yang, Y., Sun, H., Wan, Y., Ma, M., Cao, T.,Huang, Y., Edison, E., "The Need to Use Translation Software in the Classroom: Perception of Chinese International Engineering College Students in Language Learning," Journal of Engineering Research and Reports, 25(11):149-157 (2023).

[9]Chang, Y., Wang, X., Xue, M., Liu, Y., Jiang, F., "Improving Language Translation Using the Hidden Markov Model," Computers, Materials & Continua, 67(3): 3921-3931 (2021).

[10]Abdullah, A., "Undergraduate EFL Learners' Use and Acceptance of Mobile-Assisted Language Learning: A Structural Equation Modeling Approach," World Journal of English Language, 13(3): 253-253 (2023).

[11]Hossain, K., Steinfield, C., Medeiros, N., Kichuk, D., Chowdhury-M-H, H., Mannan-S, M., Ahmed-S-M, Z., "Subscription to e-resource database jstor digital library: A usage statistics of university of liberal arts bangladesh library," International Journal of Information Dissemination and Technology, 12(2): 43-46 (2022).

[12]Ren,Y., Ding,L., Qi,W., "Technical analysis of Uni-APP mobile application development," Electronic Technology and Software Engineering, (3): 41-44 (2023).

[13]Yu, J., Wu, D.X., "A rapid web development framework based on SpringBoot," Computer Programming Skills & Maintenance, (9): 31-33 (2021).

[14]Chen, F., Cai, J., Xiang, T., Liao, X., "Practical cloud storage auditing using serverless computing," Science China Information Sciences, 67(3): 132102 (2023).

[15]Oraskari, J., Törmä, S., Yalcinkaya, M., Singh, V., Beetz, J., "Sharing Linked Building Data in a Peer-to-Peer Network: ifcOWL Meets InterPlanetary File System," Journal of Computing in Civil Engineering, 38(1): 04023037 (2024).

[16]Simon, B., "Microsoft's Radius and the future of cloud-native development," https://www.infoworld.com. (2023).