



E-ISSN : 2988-585X (Online)

Journal of Elektronik Sistem Informasi (JESII)

Volume 2 No 2 Desember 2024

DOI : 10.31848/jesii.xxxx.xxxx

Implementing Microservices Architecture in Student Competition Website using Web Service Method

Hilman Nuril Hadi¹, Syntia Widayuningtias P. Listio², Bagus Kristomoyo Kristanto³

^{1,2,3}Department of Informatics, Sekolah Tinggi Informatika & Komputer Indonesia, Indonesia

Article Info

Article history:

Received Dec 5, 24

Revised Dec 30, 24

Accepted Dec 31, 24

Keywords:

Microservices

Website Competition

Web Service

ABSTRACT

Effective management of a student competition website is crucial in the contemporary digital landscape, as it can support a substantial number of participants and users with optimal performance. As the competition organizer, APTISI Jawa Timur Province often use a website as the platform for spreading information regarding these competitions. Recent advancements in these competitions encompass the introduction of additional levels and categories, participant administration functionalities, payment systems, work submissions, competition webinars, and the issuance of competition certificates. The competition website development team encounters difficulties in sustaining the website's features and quality. The swift advancement of information technology demands scalable and adaptable web solutions, especially for platforms that handle dynamic and intricate capabilities like student competition websites. This paper establishes the deployment of a microservices architecture on the student competition website (lokreatif.org) utilizing web service methodologies. The major goal is to improve the system's modularity, scalability, and maintainability by breaking the application into independently deployable services. Implementing microservices architecture in student competition websites allows each key function to be implemented as an independent service. This method not only improves the efficiency of development and maintenance but also facilitates more distributed development teams.

Corresponding Author :

Hilman Nuril Hadi,

Department of Informatics, Sekolah Tinggi Informatika & Komputer Indonesia.

Jln. Raya Tidar No.100, Karang Besuki, Kec. Sukun, Kota Malang, Jawa Timur, Indonesia. 65146

Email: hilman@stiki.ac.id

1. INTRODUCTION

In today's digital era, effective management of a student competition website is crucial as it can accommodate a large number of participants and users with optimal performance [1][2]. Efficient design and website management guarantee the swift and smooth processing of competition information, registration, and submissions, free from technical issues, thereby enhancing user experience and fostering trust in the organizers. Additionally, scalability allows the website competition to grow with increased participation and competition complexity while maintaining high service quality. Moreover, an efficient and scalable website also supports better data management, participation analysis, and

accurate and timely result reporting, all of which contribute to the success and reputation of the competition among students and the public [3].

Association of Indonesian Private Universities (APTISI) East Java Province, in collaboration with related universities, annually organizes student competitions for university students across Indonesia, covering various fields with different requirements. APTISI typically utilizes a website as the medium for disseminating information about these competitions. There have been several developments in these competitions, including the addition of new levels and categories, participant management features, payment systems, submission of works, competition webinars, and distribution of competition certificates. Additionally, the competition website development team faces challenges in maintaining the features and quality of the website used [4]. The features and quality of the website have an impact on the quality of information dissemination services, which in turn affects the satisfaction level of participants who access the information. Meanwhile, managing the quality of the website becomes complex if the management aims to maintain the website's performance and user comfort. Therefore, the competition website usually needs continuous development throughout its lifecycle to meet the demands of the competition organizers.

As technology advances, the need for systems capable of efficiently and rapidly handling diverse requests is becoming more critical. One of the emerging popular approaches in system development is the microservices architecture. This architecture facilitates the development of applications in a more modular manner, allowing for the independent development, testing, and deployment of each service or component [5]. The implementation of microservices architecture is highly relevant for various applications, including student competition websites, which require high scalability and flexibility to manage multiple functions, such as registration, participant data management, and competition evaluation.

The adoption of microservices architecture has gained significant attention in recent years due to its ability to address the limitations of traditional monolithic systems. Microservices architecture involves designing software applications as a suite of independently deployable services, each running in its own process and communicating with lightweight mechanisms, often HTTP resource APIs [7][8][9]. This modularity facilitates continuous deployment and scalability, making it an ideal solution for complex and dynamic web applications, such as student competition websites that handle fluctuating workloads and diverse functionalities. Furthermore, microservices enable more agile development processes, allowing teams to deploy new features and updates more rapidly without disrupting the entire system.

Several studies have demonstrated the effectiveness of microservices architecture in educational and competitive environments. According to Bogner et al., they examined the use of microservices in online platforms and found that they significantly enhance performance and fault tolerance [6]. These findings are particularly relevant for student competition websites, which require robust and scalable architectures to support activities such as participant registration, submission handling, and real-time scoring. Implementing microservices can ensure that these platforms remain reliable and responsive, even under peak usage conditions.

To meet customer demand, a system typically requires continuous development throughout its life cycle. Considering the challenges in managing competition websites, the need for microservices architecture in this context becomes increasingly relevant as the complexity and scale of the competitions grow each year. Microservices architecture enables the independent development and management of website components, enabling the development, management, and deployment of services like registration, participant management, work submission, and evaluation [6]. This increases flexibility and speed in development, allowing the team to update or fix one service without disrupting the entire system. Additionally, the microservices architecture enhances scalability by enabling each service to scale independently based on its workload requirements. Thus, when there is a surge in registrations or submissions, the related services can easily adjust their capacity. This architecture also enhances the system's resilience and reliability, as a failure in one service does not cause the entire system to fail. In the context of managing competition websites, adopting microservices architecture helps ensure that the system remains responsive, efficient, and capable of handling the growth and complexity of the competitions more effectively.

Student competition websites can develop each core function as a separate service by implementing microservices architecture. For instance, we can develop and manage the registration service, submission service, and scoring service independently. This approach not only enhances the

efficiency of development and maintenance but also enables more distributed development teams. Furthermore, the microservices architecture enables independent scaling of each service to manage high workloads, ensuring the competition website maintains responsiveness and reliability despite an increase in participants and visitors. This article delves into the application of microservices architecture and elucidates how APTISI implemented features on the competition website.

2. METHOD

This article employs an experimental approach to develop a student competition website, utilizing a microservices architecture and web service methods. Microservices architecture is a method for creating big apps by segmenting them into smaller, independent services. Every service can communicate with other services and has an easy-to-understand API. Increased system robustness and scalability, as well as ease of development, testing, and deployment, are some advantages of this strategy[10]. The development process starts with a requirement analysis to identify the key features required by users and competition organizers (APTISI). The first step involves gathering detailed requirements from stakeholders, including students, faculty, and administrators, to understand the functionalities needed for the competition platform. This step identifies and prioritizes all necessary features, including user registration, submission handling, judging, and result announcements. We utilize techniques like interviews, surveys, and requirement workshops to gather comprehensive and accurate requirements. We differentiate the features developed based on the perspective of the actors. The features accessible to the general users (without login) are explained in Table 1.

Table 1. Summary of Competition Website Features (without user login)

NO	Site Name	Url
1	Homepage Competition Website	https://lokreatif.org/
2	Homepage Nasional Level	https://lokreatif.org/national
3	Homepage Internasional Level	https://lokreatif.org/international
4	Competition Schedule	https://lokreatif.org/national/jadwal
5	Finalist Announcement Page	https://lokreatif.org/national/finalis
6	Winner Announcement Page	https://lokreatif.org/national/juara
7	Judges Information Page	https://lokreatif.org/national/juri
8	Download Asset and Information Page	https://lokreatif.org/national/unduh
9	Participant Statistics National Level	https://lokreatif.org/national/statistik
10	Participant Statistics International Level	https://lokreatif.org/international/statistik
11	Registration Page National Level	https://lokreatif.org/national/register
12	Registration Page International Level	https://lokreatif.org/international/register

The design of the microservices architecture involves dividing the system into independent services including registration, participant management, work submission, event management and competition data management, described in Figure 1. The implementation of microservices architecture for the student competition website follows. Once the requirements are established, the design phase begins, focusing on decomposing the system into distinct microservices. Each microservice is responsible for a specific functionality, such as user management, file storage, or scoring system. The design process leverages domain-driven design to ensure each microservice aligns with a specific domain context, which simplifies development and maintenance.

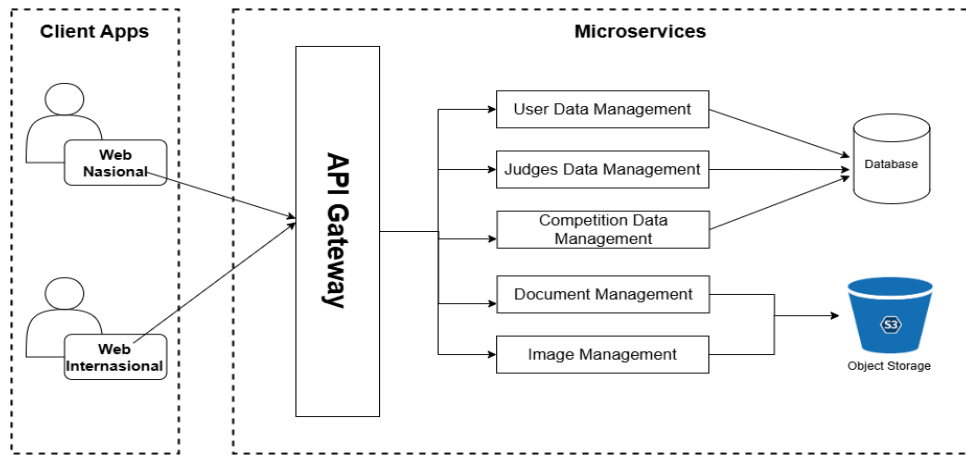


Figure 1. The Microservice Architecture of Student Competition Website

Each service is developed separately using web service methods to ensure both interoperability and flexibility. The implementation is conducted iteratively with regular testing to ensure each service operates correctly and meets user requirements. After the development phase, performance and scalability tests are conducted to assess the system's ability to manage user surges and the increasing complexity of the competition. The experimental results are then analyzed to evaluate the effectiveness of the microservices architecture in enhancing the efficiency, scalability, and reliability of the student competition website. The implementation phase involves coding, implementing user interface and integrating the microservices using an iterative and incremental approach. Each microservice is developed independently, tested, and deployed in a staging environment to ensure functionality and reliability before integration. Continuous integration/continuous deployment (CI/CD) pipelines are set up to automate the testing and deployment processes, ensuring that new features and updates can be released quickly and safely.

3. RESULT AND DISCUSSION

This research produces a website-based application that has the name lokreatif.org. The implementation results of the code and the interface of the student competition website show a significant improvement in functionality and user experience. The homepage of the student competition website is represented in Figure 2. The page is divided into two main pieces of information, the left part is for entering national level competition and the right part is for entering international level competition.

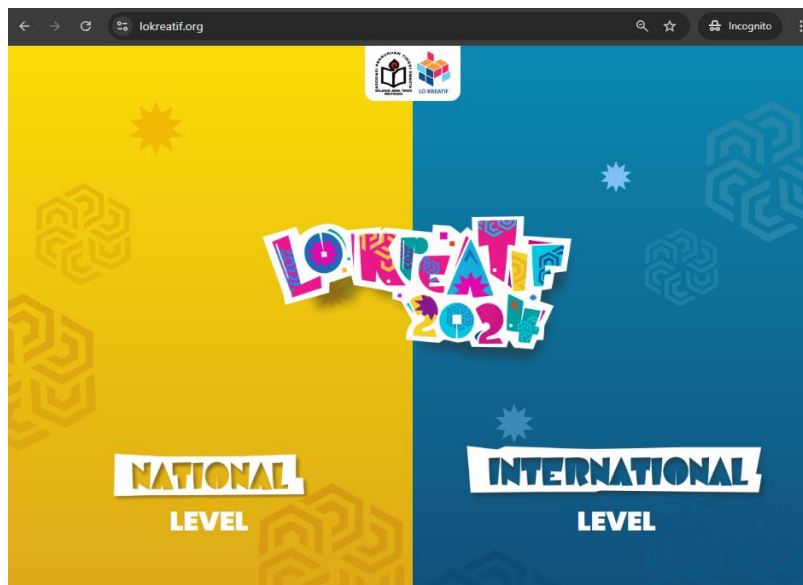


Figure 2. Homepage Competition Website

In Addition, the homepage of the national level student competition website is represented in Figure 3. The page contains some information related to national level competitions such as registration schedules, registration instructions, participant statistics, activity information and others.

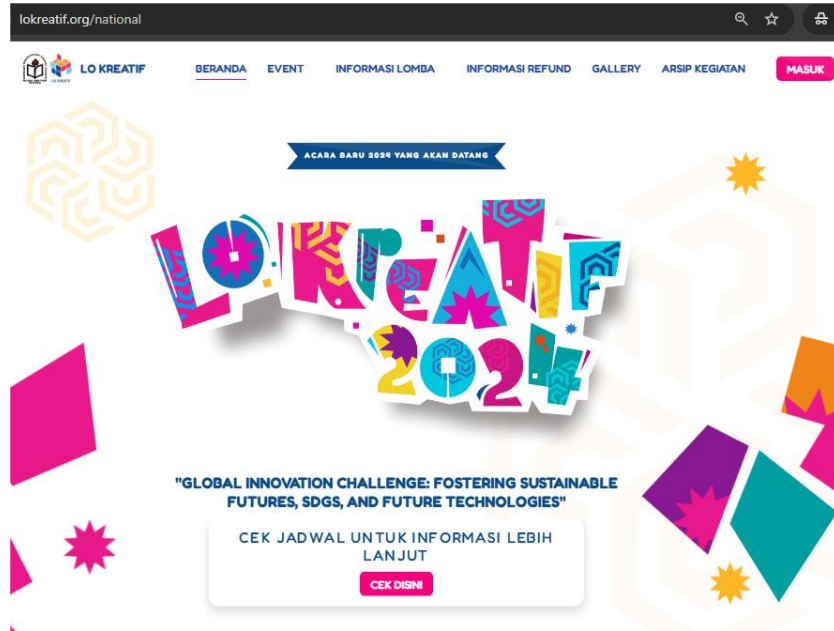


Figure 3. Homepage Competition Website for National Level

In addition, another important page on the competition website is the participant statistics page, described at Figure 4. This page contains some information on the number of participants, the distribution of participants and monitoring the status of the registration process

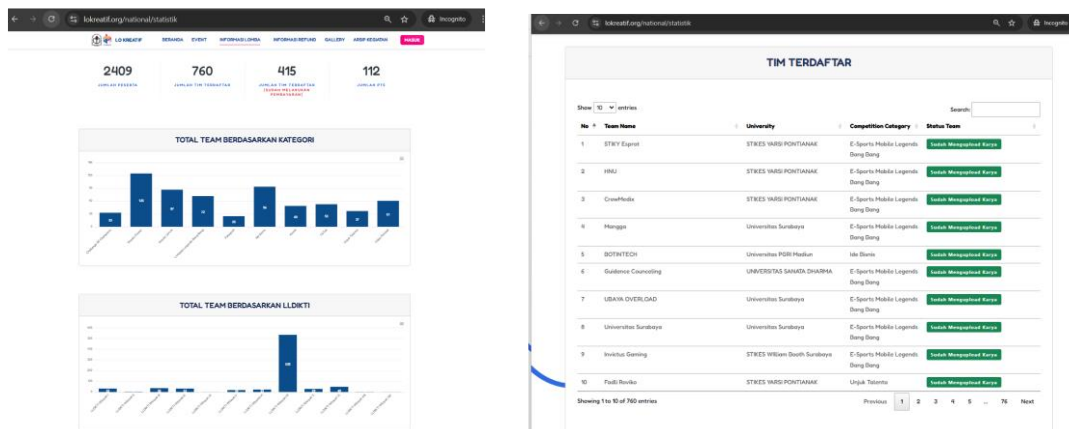


Figure 4. Statistics Page for National Level

To enhance the system's robustness, comprehensive performance and load testing were performed. The tests replicated peak usage scenarios, including concurrent submissions during competition deadlines, to assess the system's capacity to manage high traffic. The findings demonstrated

that the modular structure of the microservices architecture facilitated the autonomous scalability of resource-intensive services, such as the submission service, without affecting other components like registration or scoring.

Moreover, dependability and fault tolerance were primary focal points during the installation. A fault-isolation technique was implemented to guarantee that failures in one service did not compromise the functionality of others. For instance, if the submission service had downtime, essential services such as registration and event administration remained operational without interruption. The system presently employs a messaging-based communication method for service integration; however, the implementation of technologies such as REST APIs could significantly improve interoperability and scalability in the future[11]. REST APIs, due to their simplicity and standardized communication protocols, facilitate the integration of third-party solutions and promote more flexible interactions between services[12][13]. Implementing this technology could significantly benefit the platform, especially as it becomes more complicated and requires support for additional features or external connectors.

4. CONCLUSION

The microservices architecture on the student competition website has provided significant benefits. This architecture enables the development and deployment of services independently, enhancing system flexibility and scalability. Developers can independently develop, test, and deploy each separate service, thereby accelerating the development cycle and adapting new features to user needs. For example, when there is a data correction for national-level competition participants, developers can make the correction and deploy it without affecting other features, such as those for international-level competition participants. Additionally, the microservices architecture facilitates workload management by allocating resources efficiently, ultimately improving the performance and reliability of the website.

ACKNOWLEDGEMENTS

Thank you to all parties who have supported this research so that it can be completed properly. Especially, for the parents who have always supported and guided me in this research. I'm very grateful.

REFERENCES

- [1] D. Yulianto and T. Ismail, "Analisis Website Program Kreativitas Mahasiswa (PKM Center) Universitas Ahmad Dahlan Menggunakan WebQual 4.0," *MATRIK: Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 20, no. 2, pp. 325–334, May 2021, doi: 10.30812/matrik.v20i2.1098.
- [2] Y. Yudhanto and K. S. Andien, "Rancang Bangun Sistem Informasi Lomba Start-Up Berbasis Web," *IJAI (Indonesian Jopurnal of Applied Informatics)*, vol. 7, no. 1, pp. 1–12, 2022.
- [3] D. Yulianto, A. R. C. Baswara, L. Alhawariy, M. I. Prasasti, and G. A. Hariadi, "Development of Information and Management System of Student Competition Groups through User-Centered Design Approach," vol. 9, no. 1, Apr. 2023.
- [4] D. A. S. Sugiono, A. Aditya, H. N. Hadi, and F. E. Purwiantono, "Analisis Usability Testing Guna Meningkatkan Efisiensi Kerja (Studi Kasus: Website Lo Kreatif)," 2022. [Online]. Available: www.tegalkota.go.id
- [5] S. A. Karimah, H. H. Latif, and S. Prabowo, "Analisis Performansi Layanan Web Menggunakan Arsitektur Microservice Dan Monolitik," *KESATRIA: Jurnal Penerapan Sistem Informasi (Komputer & Manajemen)*, vol. 4, no. 4, pp. 830–840, 2023.
- [6] J. Bogner, J. Fritzsich, S. Wagner, and A. Zimmermann, "Microservices in Industry: Insights into Technologies, Characteristics, and Software Quality," in *Proceedings - 2019 IEEE International Conference on Software Architecture - Companion, ICSA-C 2019*, Institute of Electrical and Electronics Engineers Inc., May 2019, pp. 187–195. doi: 10.1109/ICSA-C.2019.00041.

- [7] R. T. Adek, S. Mulaesyi, Z. Yunizar, and M. Fauzan, "IMPLEMENTASI HPACK DAN GZIP PADA OPTIMASI APLIKASI DENGAN ARSITEKTUR MICROSERVICE," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 11, no. 2, pp. 421–428, 2019.
- [8] S. Atmojo, R. Utami, S. Dewi, and N. Widhiyanta, "Implementasi Sistem-informasi Desa Berbasis Arsitektur Microservices," *SMATIKA JURNAL*, vol. 12, no. 01, pp. 55–66, Jun. 2022, doi: 10.32664/smatika.v12i01.658.
- [9] S. Baškarada, V. Nguyen, and A. Koronios, "Architecting Microservices: Practical Opportunities and Challenges," *Journal of Computer Information Systems*, vol. 60, no. 5, pp. 428–436, Sep. 2020, doi: 10.1080/08874417.2018.1520056.
- [10] N. Miracle dan R. Puspita Wibawa, "Implementasi Arsitektur Microservice pada Simulasi Konsolidasi Bisnis Website Menggunakan Metode Extreme Programming." *Jurnal Informatika dan Ilmu Komputer (JINACS)*. vol. 6, no. 2, hlm. 493-500, Juli 2024.
- [11] A. A. Prayogi, M. Niswar, Indrabayu, and M. Rijal, "Design and Implementation of REST API for Academic Information System," in *IOP Conference Series: Materials Science and Engineering*, Institute of Physics Publishing, Jul. 2020. doi: 10.1088/1757-899X/875/1/012047
- [12] I. Ahmad, E. Suwardi, R. I. Borman, Asmawati, F. Rossi, and Y. Jusman, "Implementation of RESTful API Web Services Architecture in Takeaway Application Development," in *International Conference on Electronic and Electrical Engineering and Intelligent System (ICE3IS)*, 2021, pp. 132–137.
- [13] B. Redika, W. Putra, and E. Mailoa, "Implementation Of Microservices In Fintech Applications Using Expressjs Framework," *Eduvest-Journal of Universal Studies*, vol. 1, no. 7, pp. 559–567, 2021, [Online]. Available: <https://greenvest.co.id/>