Design of decision support system service in the Space Science Center using microservices approach

Elyyani(1), Y. Andrian(2), A.Z. Utama(3), M.F. Eko Saputro(4), S.K. Fatimah(5)
1,2,3,4,5 Space Science Center – National Institute of Aeronautics and Space

Email: elyyani@brin.go.id

Abstract. Development of the Decision Support Systems (DSS) has been written up in the Space Science Center's roadmap for the next few years. The main programs have to be achieved such as related to the DSS are Space Weather, High Frequency Radio Communication, Space Debris Monitoring, and Navigation activities. Of course, accessing those DSS services separately may become a problem for users who need information as fast as possible. We proposed an information system as a portal to integrate all DSS services collecting the latest information from each main program. We applied the microservice approach for design of this Information System prototype, which will it grows up as the modern application model that is capable to handle multiple features, high scalability but still lightweight. Microservices approach is a method that breaks an application into a single-function module for optimizing its performance. In addition, usage of web service technology and REST-ful API also can support system interoperability functions. In this design we conducted several stages, use case Diagrams, design of microservices database, and prototyping frontend pages in the results. This study purposes to provide an initial system design with a detailed explanation of specification requirements so that the design of the Decision Support System Service in the Space Science Center environment can be developed as expected.

1. Introduction

Development of the Decision Support Systems (DSS) has been written up in the Space Science Center’s roadmap for the next few years. The main programs have to be achieved such as related to DSS are Space Weather, High Frequency Radio Communication, Space Debris Surveillance system Monitoring, and Navigation activities. An information system is needed as a portal for DSS services. DSSs are a sub-collection of information management systems that help planners, analyzers and managers in decision making process [1]. Several DSS services at the Space Science Center need to be integrated through an information system as an interface for users to find the information needed. The development of the information system is supported by the availability of hardware and software that functions to collect, process, store and distribute information. Decision support systems to support organizational decisions process combine in other information systems such as management information systems and provide middle managers possibility to examine problems solving by provided information and make necessary decision [1]. The Decision Support System is intended to complete the information system management in improving decision making.

![Figure 1](image_url)  
Figure 1. Problem structuring grades and using decision support system [1].
The figure 1 there are various structures in decision making, structured problems can be solved through computer systems that have routine problems and standard procedures. While the semi-structured problem is a problem between structured and unstructured, decision making in this case needs consideration from the manager who will be used in the decision-making system (DSS). With the DSS, companies can solve problems easier, both structured and unstructured.

Because of the increment types and large amount of information will make some problems in developing and maintaining the code of information system. It can have an impact on system performance and certainly be a problem in the information systems development. Microservice architecture is used to anticipate various problems during application development in the future. The microservice architectural style is an approach to developing a single application as a suite of small services, each running in its own process and communicating with lightweight mechanisms. It is often as an HTTP resource API [2]. The software was divided into smaller parts which it easier for developers to develop each service without disturbing other services.

The figure 2, Google trend shows statistics for the microservice architecture usage [3].

![Google Trends for Microservice](image)

**Figure 2.** Google trends for microservice [3].

The development of the DSS service is a replica description of the information system that will be developed into an integrated service. The purpose of this work makes large system easier to manage and better for scalability. This design also easier to maintain and to deploy because in microservices each service can be deployed independently.

2. Data and Method

2.1. Data

The data are from four DSS in Space Science Center in the form of internal data, research data / information generated from DSS group services are Space Weather, High Frequency Radio Communication, Space Debris Surveillance System Monitoring, and Navigation activities.
2.2. Method
The architecture for application development can be adjusted based on needs such as how big the application is to be built. The development of applications that are getting bigger and the number of accessor is increased will affect the performance of the application. Currently the monolithic approach has shifted to a distributed approach, where applications are divided into small parts that function specifically (high cohesion) and do not depend on other program components (loose coupling), with the development of services through an API interface (Application Interface) [4].

Monolithic architecture is an application with a single codebase that includes multiple services. These services communicate with external systems or consumers via different interfaces like Web services, HTML pages, or REST API [5]. Figure 4 displays monolithic and microservice architecture different. Monolithic architecture consists of three main layers: user interface, business logic and data interface. User interface that is responsible for interface of software and its interaction with user. Business logics responsible for interactions between business objects, and methods in which they are handled, and updated. The data layer defines the rules and methods used for storing, retrieving, processing, and transferring the data. It includes a database and a database management system [6].

Monolithic architecture is the application development process where all the components to build the application are stored into a single unit, have a single codebase, built in single application and single database which has a centralized architecture with uniform technology. In this case the front-end application and the back-end application are in one application. Monolithic systems are created in the same code base, divided into multiple modules according to the business or technical features of the system [7]. Microservice architecture divides the system into smaller units called service and each service handles only one part of the software system that each other communicate [7].

![Image](image-url)
One of the most important parts of the architecture is performance, however there are advantages and disadvantages of the used architecture [8], displayed on the following Table 1.

| Pros          | Microservice                          |
|---------------|---------------------------------------|
| Easy to develop | Easy maintenance (easier to understand by a developer, each core functionality is a separated module) |
| Simple to deploy | Reliability (microservice fault affects only that microservice alone) |
|               | Availability (redeploy a new version of a microservice requires little downtime) |
|               | Easy to scale                         |

| Cons          | Microservice                          |
|---------------|---------------------------------------|
| Complex maintenance | More complex deployment               |
| Reliability (one fault can bring down the entire application) | Autonomy (it is a positive aspect, however to achieve this along with data integration is a huge challenge) |
| Availability (redeploy the entire application on each update) |                                   |
| Hard to scale |                                      |

Research on the microservice approach has also been widely carried out, including being used in the development of information system designs to show an increase in the quality of resilience, resilience means being able to adapt to changes and errors/damage to infrastructure [9]. The development of research and technology, the application of microservice technology is also used for the development of mobile learning through the lumen framework [10].

3. Result

Development of the DSS service is a web portal as an interface between the user and all DSS services at Space Science Center. In the development of information systems, it is necessary to first describe the scope and limitations of a system including other systems that interact with it, described in context diagrams, use case diagrams and system architecture to be built.
3.1 Context diagram

Context diagrams show a system, as a whole and its inputs and outputs from/to external factors. This context diagram describes the interaction of the system with external entities such as users and administrators. In this case the admin has full authority over the system and data management. Context diagrams show a system, as a whole and its inputs and outputs from/to external factors. This context diagram will describe the interaction of the system with external entities such as users and administrators.

In this case the admin has full authority over the system and data management, figure 5.

Figure 5. Context diagram for DSS services.

3.2. Use case diagram

The use case diagrams are typically developed in the early stage of development and people often apply use case modeling for the following purposes: determine of the context system and requirements system, validate a systems architecture. Use case diagram is a diagram that describes the relationship between actors and the system. Use case model is the interaction and relationship of individual use cases [11]. In this case there are administrators and users as actors on the system. Figure 6, backend pages are used by administrators to manage information. The frontend is used to provide user-accessible information such as DSS service information on space weather, space debris surveillance system and Frequency radio communication.

Figure 6. Use case diagram for DSS service.
3.3. System architecture
At this stage it will display design architecture on the information system development based on a microservice architecture consisting of the database system process, design architecture frontend and backend service and requirement specification on the application.

3.3.1. Database system process on Space Science Center DSS information system.

The picture above (figure 7) is a database design that uses the concept of microservice architecture. In this architecture the application will be divided into several smaller services as DSS of Space Weather, Frequency Radio Communications, Space Debris system Monitoring, and Navigation so that the services are connected even though they use different applications. In terms of system management and development, the development can freely develop each service without disturbing other services so that if there is a change it will be easier.

Database design on DSS’s Space Science Center uses database per service type, when each service uses its own database such as the space weather database, space debris surveillance system database and the HF radio Communication database.

3.3.2. Design frontend and backend service
The following is the result of the database architecture design for the frontend service and backend services. Between the frontend and backend services are connected with Rest as the architecture used in the API, with the data provided by the Rest server in the form of Json format, where the database processing uses MongoDB.
Figure 8. Architecture design frontend and backend service.

Figure 8 illustrates the frontend of the application calls multiple service. Each of these services have their separate codebases and if there is a need to communicate between them, the communication is done through the APIs these services provide.

3.3.3. Requirement specification for DSS service application process

System requirements are very important in building an information system, here are the minimum requirements for the hardware and software, figure 9 and figure 10.

| Module | Software requirements |
|--------|-----------------------|
| Application development tools for Frontend | Angular CLI. |
| Application development tools for Backend | Node Express ver 4.17.1 |
| Database | MongoDB ver 4.4.5. |

Figure 9. Hardware requirement.

Figure 10. Software requirement.
The requirements software is Angular, for application development tools for Frontend. Angular is a platform and framework for building single-page client applications using HTML and TypeScript. NodeJS for running web pages, file manipulation, database manipulation and user interaction manipulation. Node Express, for backend web application framework. MongoDB, for database system (MongoDB is a source-available cross-platform document-oriented database program). MongoDB is a NoSQL database system, which is a document-oriented and open source for querying and indexing [12].

3.3.4. Application display design
This stage will display the page view from the frontend and backend. Figure 11, backend page for managing various information services on the DSS Services Space Science Center. Backend pages are developed using the Angular CLI.

![Figure 11. Backend page DSS service](image)

In the figure 12, frontend page that can be used for users to access DSS service information through the features provided. On this Frontend page only display for one of the DSS services, for example DSS High Frequency radio communication. Globally this information system provides portal links related to DSS services, as well as the latest information from each DSS.
4. Conclusion
Currently the DSS (Decision Support System) service is being developed at the Space Science Center, to anticipate future technological changes and to maintain application performance, this microservice architecture is very appropriate used to develop applications. Microservice style architecture become the standard in building dynamic systems.

Microservice architecture is more efficient if an application has to handle a bigger number of requests. It has a lot of advantages that allow to build a high-quality software, which is easy to scale, more reliable and in long term more convenient for maintenance. With microservices, we are also able to adopt technology more quickly.

Acknowledgements
The author would also like to thank the Space Science Research Center, Indonesian National Institute of Aeronautics and Space (LAPAN) for funding and facilitating this research in 2021. The authors' contributions were as follows: Elyyani designed the research and wrote the manuscript; Elyyani, Andrian, and Utama designed the microservices architecture and analyzed data; Utama, Eko Saputro, and Fatimah developed the microservices application by using mentioned framework. Elyyani, Andrian, and Utama contributed as the main contributor to this paper and had primary responsibility for the final content of the manuscript. All authors read and approved the final paper. Authors also want to thanks the editor and the reviewer for helping evaluate the manuscript.

References
[1] Khodashahri NG Sarabi MMH 2013 Decision Support System Singaporean journal of business economics and management studies Vol 1 No 6
[2] M. d. J. L. Folwer, "Microservices," Available: https://martinfowler.com/microservices/. [Accessed 2 02 2021]
[3] https://trends.google.pl/trends/explore?date=today%20-5-y&q=microservices
[4] Newman S 2015 Building Microservices: designing fine-grained system First Edition Beijing Sebastopol CA: O’Reilly Media Inc
[5] Al-Debagy O and Martinek P 2018 A Comparative Review of Microservice and Monolithic Architectures CINTI 2018 18th IEEE International Symposium on Computational Intelligence and Informatics Nov. 21-22, 2018 Budapest, Hungary
[6] https://jelvix.com/blog/monolith-vs-microservices-architecture [Accessed 3 08 2021]
[7] https://www.hitechnectar.com/blogs/microservices-vs-monolithic/ [Accessed 4 06 2021]
[8] Zabierowski W 2020 The Comparison of Microservice Dan Monolithic Architecture. MEMSTECH 2020 22-26 April, 2020 Lviv UKRAINE Conference Paper April 2020 DOI: 10.1109/MEMSTECH49584.2020.9109514
[9] Suryotrisongko H 2017 Arsitektur Microservice untuk Resiliensi Sistem Informasi Jurnal Sisfo Vol 06 No 02 (2017) 235–250
[10] Sandiang M, Kasenda S and Purnama J 2018 Implementasi Teknologi Mikroservice pada Pengembangan Mobile Learning Journal of Applied Informatics and Computing (JAIC) Vol 2 No 2 Desember 2018, pp 63–66 e-ISSN: 2548-6861
[11] Kendal Kenneth E Kendall Julie E 2010 System Analysis and Design 8th Edition Upper Saddle River New Jersey: Prentice Hall
[12] Chen R, Li S and Li Z 2017 From Monolith to Microservices: A dataflow-driven approach in 24th Asia-Pacific Software Engineering Conference