

Applied Research Adventurehub

By Arenco Meevissen



7 oktober 2022

**Revision Table**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | change(s) |
| 1.0 | 7/10/2022 | Arenco Meevissen | Initial document |
| 1.1 | 2/11/2022 | Arenco Meevissen | Added answer to all questions, added conclusion and recommendation |
| 1.1.1 | 4/11/2022 | Arenco Meevissen | Changed title to include project name |

Contents

[Main question: What relation data storage system provides the best trade-off between the structure and security of the application for my individual project? 3](#_Toc120264517)

[Sub question 3](#_Toc120264518)

[What is the project structure that I need? 3](#_Toc120264519)

[What security parts of my application are relevant for the data storage system? 3](#_Toc120264520)

[How much storage is needed in the next five years? 3](#_Toc120264521)

[What data storage systems can be connected to my application? 4](#_Toc120264522)

[How do a MySQL, Microsoft SQL server and Postgress database interact with my structure and security, regarding security and storage? 4](#_Toc120264523)

[Conclusion & Recommendation 4](#_Toc120264524)

[References 5](#_Toc120264525)

# Main question: What relation data storage system provides the best trade-off between the structure and security of the application for my individual project?

# Sub question

## What is the project structure that I need?

With the methodologies: IT architecture sketching, SWOT analysis, Design pattern research

First, I used Design pattern research to search for standard structures.  
Then by using the IT architecture sketching I made an overview with what I currently have and what I plan to do. Then I used SWOT analysis for what the individual parts of my architecture I need and what can be changed.

I looked at what are the commonly used structures the most used ones are separating per feature and then by layer (GeeksforGeeks, 2021; Developing with Spring Boot, z.d.). Then I used these 2 methods and looked at what worked for me better and can be extended better and what their strength and weaknesses are following Vivekananda the simplicity, maintainability, flexibility, and scalability are all maintained but a weakness on this is that with more layers you have more risk for things to breakdown or data to get lost (2015). While following Griffin its main strength lays in the specialization and productivity and their weaknesses lay more in the management and unit coordination (2019) so I chose the layered approach because I think this will help me more with maintaining a clear architecture.

## What security parts of my application are relevant for the data storage system?

With the methodologies: SWOT analysis, Security test, Design pattern research, Benchmark test

First, I used SWOT analysis together with the Design pattern research for what the individual parts of my architecture I need to connect to security and what to the database. The security test was next to see where the weaknesses are in the security. With the last one being the benchmark test to see how I did it to others.

Because I already have chosen an architecture that I want I look at how I can introduce the security together with the data storage and I did this by introducing a security layer which I can keep it simple, flexible in case something doesn’t need as much security and its maintainability.   
My architecture compared to others and how I will implement this is I think I did well but there can be some more division if you look at the official site. (Spring Security Architecture, z.d.)

## How much storage is needed in the next five years?

With the methodologies: SWOT analysis, Design pattern research, Benchmark test

First the SWOT analysis and Design pattern research to determine how can I structure best the needed data while keeping it as small as possible. Then I did the Benchmark test for comparison to other similar data structures.

I first looked if there was already a similar project like what I did but not really the most closed one I consider is sribblehub.com which has a forum and a website where users can create their own stories. I calculated an average size I think this website uses and look into possible tables this website has and what tables I want to have and what the similarities are between my project and this website. While I think the used data type for storing the chapter is between text and medium text so I will go for the bigger one for “max” value it could possibly be so 16MB per chapter and a story can have multiple of these chapters I think their “max size” for their story related stuff (chapters, glossary, introduction/index) is for a normal story (around 20-30 chapters) between like the 320 and 480 MB and this site has a lot of these and you can even add images into the story and saving images is also a huge size compared to only text so I thought I will only do text because with that I can save some data. (Estimate the Size of a Database - SQL Server, 2021b) and (Welch, 2016) and (Understanding File Types, Bit Depth, Image Data Size. With calculators for image size and KB, MB and GB conversion, z.d.)

## What data storage systems can be connected to my application?

With the methodologies: Available product analysis, best good and bad practices, and Community research.  
By using Available product analysis I searched for what are the databases considered that you can connect to. Then by using best good and bad practices together with community research to see what I can consider to be used by me.

I used a website to see what databases I connect to which is roseindia (Hibernate 4 Supported Databases List, z.d.) which I consider useful because of the extensive tutorials they have on their website and the comments on their YouTube channel. So, I looked into what databases I already knew and then to the less familiar ones where I at last chosen 3 databases base on amount of search result you can get, and examples given.

## How do a MySQL, Microsoft SQL server and Postgress database interact with my structure and security, regarding security and storage?

With the methodologies: A/B testing, Guideline conformity analysis, multi-criteria decision making, Decomposition

With A/B testing I will look at what are the differences between the 3 chosen databases. Meanwhile with Guideline conformity analysis I look at what gives the most reliability, security and then the max storage size to rank them based on those 3 values. Then I perform multi-criteria decision making on it and then I do Decomposition to break it down in smaller parts.

I looked at all 3 and I think in the order of best to least is MySQL, Postgress and then Microsoft management (D.B-Engines ranking). While MySQL hasn’t the best security, I think I rather have a more stable architecture and compromise a bit on the security the database has itself. While Postgres has the best security, I need to change my architecture to use it in the best possible way.

# Conclusion & Recommendation

In conclusion I think will choose the MySQL database for its large community, the architecture I can keep and although not the best security it will be enough, and it has enough size for my project.

The recommended actions are to implement the chosen architecture (layered design) and as fast as possible the security and data storage structure after it with the chosen database being MySQL.

# References

*DB-Engines Ranking*. (n.d.). DB-Engines. Retrieved October 7, 2022, from https://db-engines.com/en/ranking

*Hibernate 4 Supported Databases List*. (z.d.). Retrieved op 2 November 2022, from https://www.roseindia.net/hibernate/hibernate4/hibernate4databases.shtml

GeeksforGeeks. (2021, 12 July). *Spring Boot - Code Structure*. https://www.geeksforgeeks.org/spring-boot-code-structure/

*Developing with Spring Boot*. (z.d.). Retrieved on 2 November 2022, from https://docs.spring.io/spring-boot/docs/current/reference/html/using.html

Griffin, D. (2019, 25 January). *Functional Organizational Structure Advantages*. Small Business - Chron.com. https://smallbusiness.chron.com/functional-organizational-structure-advantages-3721.html

Vivekananda, G. (2015, October). *Critical analysis of cross-layer approach*. researchgate.net. https://www.researchgate.net/publication/291075332\_Critical\_analysis\_of\_Cross-layer\_approach

*Spring Security Architecture*. (z.d.). spring. Retrieved on 2 November 2022, from https://spring.io/guides/topicals/spring-security-architecture

*Estimate the Size of a Database - SQL Server*. (2021b, May 25). Microsoft Learn. https://learn.microsoft.com/en-us/sql/relational-databases/databases/estimate-the-size-of-a-database?view=sql-server-ver16

Welch, A. J. (2016, 1 January). *Understanding Storage Sizes for MySQL TEXT Data Types*. Chartio. https://chartio.com/resources/tutorials/understanding-strorage-sizes-for-mysql-text-data-types/

*Understanding File Types, Bit Depth, Image Data Size. With calculators for image size and KB, MB and GB conversion*. (z.d.). Retrieved on 2 November 2022, from https://www.scantips.com/basics1d.html