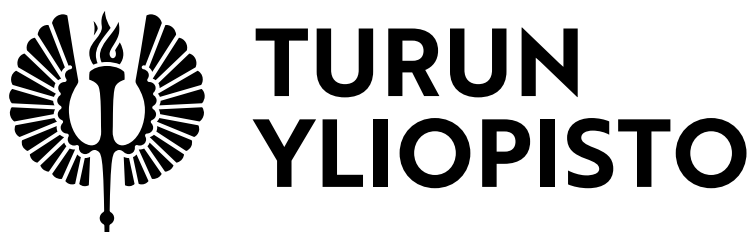


TURUN YLIOPISTO — UNIVERSITY OF TURKU
FACULTY OF TECHNOLOGY



Research Proposal

Master's Thesis in Technology (DTEK1002)

Florian Dejonckheere

fwdejo@utu.fi

January 26, 2024

Contents

| | | |
|---|----------------------|---|
| 1 | Abstract | 2 |
| 2 | Aims and goals | 3 |
| 3 | Research questions | 4 |
| 4 | Research methodology | 5 |
| 5 | Contents | 6 |
| | References | 7 |

1 Abstract

Working title: *A Framework for Writing Distributed Applications using the Modular Monolith Software Architecture*

Modern applications designed to run in the cloud are often programmed as a set of microservices: a decomposition of the application based on logical boundaries in the domain logic. This separation of concerns brings along a number of advantages such as scalability, fault isolation and independent deployment, over more traditional software development practices such as monolithic codebases. The adoption of a microservice-based architecture also introduces a number of challenges for the developers. It requires a more thorough understand of the domain logic, and increases the complexity of the codebase.

In this thesis, we investigate and design a software architecture for deploying and managing monolithic codebases in dynamic languages as a set of microservices. The architecture transparently decomposes the application into separate microservices based on logical boundaries, and deploys them in a distributed fashion. Communication between the microservices is handled entirely by the architecture, so that the developers can concentrate on the domain logic of the application. The deployed application is scaled up and down based on the current load, and the microservices are migrated between nodes to optimize resource usage. This approach takes away the overhead of developing a microservice-based architecture, while retaining the advantages of such an architecture.

The solution is designed to be language-agnostic, but we focus on the implementation for the Ruby programming language. We evaluate the architecture by implementing a proof-of-concept and comparing it to existing solutions.

2 Aims and goals

3 Research questions

Research Question 1: What defines the modular monolith architecture, and how is it different from monolithic and microservice architectures?

Research Question 2: Which challenges and opportunities arise when considering adoption of the modular monolith architecture for dynamic languages compared to statically-typed languages?

Research Question 3: How effective is the modular monolith architecture in dynamic languages, and what is its impact on overall system performance and scalability?

4 Research methodology

5 Contents

1. **Abstract**
2. **Introduction:** general introduction to the topic, and identification of challenges and obstacles related to the topic. Explanation about why and for whom the topic is relevant, and what the expected outcome of the research is. Scope of the research thesis.
3. **Background:** historical background of the topic, and theoretical background needed to understand the thesis.
4. **Related work:** relevant work and research related to the topic.
5. **Methodology:** explanation of the research methodology used in the thesis.
6. **Analysis:** analysis of the problem, and the proposed solution to the research questions. Details design decisions and implementation details.
7. **Case study:** application of the proposed solution to a real-world use case.
 - 7.1. **Background:** background information about the use case.
 - 7.2. **Evaluation:** evaluation of the proposed solution in the context of the use case.
 - 7.3. **Results:** results of the evaluation.
 - 7.4. **Discussion:** analysis and discussion of the results.
8. **Conclusion:** summary of the thesis, and answers to the research questions.
9. **Future work:** possible future work related to the topic.
10. **Bibliography**

References

- Abgaz, Y., Mccarren, A., Elger, P., Solan, D., Lapuz, N., Bivol, M., Jackson, G., Yilmaz, M., Buckley, J., & Clarke, P. (2023). Decomposition of Monolith Applications Into Microservices Architectures: A Systematic Review. *IEEE Transactions on Software Engineering, PP*, 1–32. <https://doi.org/10.1109/TSE.2023.3287297>
- Almeida, J., & Silva, A. (2020, September). Monolith Migration Complexity Tuning Through the Application of Microservices Patterns. https://doi.org/10.1007/978-3-030-58923-3_3
- Alshuqayran, N., Ali, N., & Evans, R. (2016). A Systematic Mapping Study in Microservice Architecture. *2016 IEEE 9th International Conference on Service-Oriented Computing and Applications (SOCA)*, 44–51. <https://doi.org/10.1109/SOCA.2016.15>
- Anand, V., Garg, D., Kaufmann, A., & Mace, J. (2023). Blueprint: A Toolchain for Highly-Reconfigurable Microservice Applications. *Proceedings of the 29th Symposium on Operating Systems Principles*, 482–497. <https://doi.org/10.1145/3600006.3613138>
- Andrade, B., Santos, S., & Silva, A. R. (2022). From Monolith to Microservices: Static and Dynamic Analysis Comparison. <https://doi.org/10.48550/arXiv.2204.11844>
- Bacchiani, L., Bravetti, M., Giallorenzo, S., Mauro, J., Talevi, I., & Zavattaro, G. (2021). Microservice Dynamic Architecture-Level Deployment Orchestration. *LNCS-12717*, 257–275. https://doi.org/10.1007/978-3-030-78142-2_16
- Barde, K. (2023). Modular Monoliths: Revolutionizing Software Architecture for Efficient Payment Systems in Fintech. *International Journal of Computer Trends and Technology*, 71, 20–27. <https://doi.org/10.14445/22312803/IJCTT-V71I10P103>
- Ghemawat, S., Grandl, R., Petrovic, S., Whittaker, M., Patel, P., Posva, I., & Vahdat, A. (2023). Towards Modern Development of Cloud Applications, 110–117. <https://doi.org/10.1145/3593856.3595909>
- Gonçalves, N., Faustino, D., Silva, A. R., & Portela, M. (2021). Monolith Modularization Towards Microservices: Refactoring and Performance Trade-offs. *2021 IEEE*

- 18th International Conference on Software Architecture Companion (ICSA-C)*, 1–8. <https://doi.org/10.1109/ICSA-C52384.2021.00015>
- Kendall, S. C., Waldo, J., Wollrath, A., & Wyant, G. (1994). *A Note on Distributed Computing* (tech. rep.). Sun Microsystems, Inc. <https://doi.org/10.5555/974938>
- Su, R., & Li, X. (2024). Modular Monolith: Is This the Trend in Software Architecture? <https://doi.org/10.48550/arXiv.2401.11867>
- Villamizar, M., Garcés, O., Castro, H., Verano, M., Salamanca, L., Casallas, R., & Gil, S. (2015). Evaluating the monolithic and the microservice architecture pattern to deploy web applications in the cloud. *2015 10th Computing Colombian Conference (10CCC)*, 583–590. <https://doi.org/10.1109/ColumbianCC.2015.7333476>
- Wolfart, D., Assunção, W. K. G., da Silva, I. F., Domingos, D. C. P., Schmeing, E., Villaca, G. L. D., & Paza, D. d. N. (2021). Modernizing Legacy Systems with Microservices: A Roadmap. *Proceedings of the 25th International Conference on Evaluation and Assessment in Software Engineering*, 149–159. <https://doi.org/10.1145/3463274.3463334>