

A Catalog of Micro Frontends Anti-patterns

Nabson Silva

Federal University of Amazonas (UFAM)
Manaus, Amazonas, Brazil
nabson.paiva@icomp.ufam.edu.br

Tayana Conte

Federal University of Amazonas (UFAM)
Manaus, Amazonas, Brazil
tayana@icomp.ufam.edu.br

Abstract

Micro frontend (MFE) is an architectural style that decomposes a monolithic frontend application into smaller, manageable, and independently deployable slices. Despite its increasing adoption, the field remains relatively underexplored, particularly in terms of identifying challenges and documenting best practices. Therefore, the goal of this Master's Thesis is to propose and evaluate an artifact that supports developers in implementing MFE architectures. We introduce a catalog of MFE anti-patterns that document common problems and practical solutions. After proposing the catalog's initial version, we conducted three empirical studies, a Personal Opinion Survey, a Controlled Experiment, and Multivocal Literature Review. After each study, we refined the catalog to produce a final version that helps developers identify, solve, and prevent problems when working with MFE architectures. The contributions of this thesis include centralized documentation of common issues and solutions when developing MFE architectures, empirical evidence on how the catalog can be used, a web application that showcases the anti-patterns and promotes collaboration within industry practitioners, and the development of MFE teaching material that instructors can integrate into software architecture curricula. We believe that the results of this work have the potential to drive significant advances in both the practice and theory of MFE, helping shape future research and improve industry adoption.

Keywords

Micro Frontends, Anti-patterns, Empirical Software Engineering, Software Architecture

1 Introduction

Software architecture focuses on software design at the highest level of abstraction [6]. At this level, architects are concerned not with specific classes or interfaces but with how the system's components, modules, or layers are integrated [15]. As a monolithic application grows, it becomes challenging to scale due to limitations like technology constraints, the necessity for vertical scaling only, and the need to reboot the entire application with each deployment [3]. To address these issues, developers are adopting the Microservice (MS) architectural style to create autonomous, distributed, and loosely coupled services [2, 4, 7]. This architecture enables teams to work independently, reducing the development time for new features. However, different teams often still need to share the same codebase for the presentation layer.

Micro Frontend (MFE) is an architectural style that extends the principles of MS to the frontend, breaking down a complex frontend application into smaller, manageable, and independently deployable

slices [9, 10]. This approach facilitates independent testing, development, and deployment of front-end components, enabling teams to work autonomously and reducing the development time for new features [5]. However, some issues faced when adopting MFEs are increased payload size, UX consistency, complex monitoring and debugging, state management, and duplicated code [10].

The objective of the thesis is **to develop an artifact to support developers when implementing and maintaining MFEs architectures**. We propose, evaluate, and evolve a catalog of MFE anti-patterns to preserve architectural integrity and assist developers in making well-informed decisions. This two-page abstract summarizes the previously published paper *Designing and Evaluating a Catalog of Micro Frontends Anti-patterns* [11], which presents the main contributions of Nabson Silva's Master's thesis, supervised by Professor Tayana Conte.

2 Research Methodology

We adapted the evidence-based methodology proposed by Mafra et al. [8] to develop and evaluate the catalog. First, we conducted a Rapid Review and found no documented MFE anti-patterns. Based on anti-patterns previously reported for MS, we then proposed an initial catalog containing 12 MFE anti-patterns.

We refined and evaluated the catalog through three complementary studies. A Personal Opinion Survey with practitioners assessed whether the proposed anti-patterns represented problems observed in practice. A controlled quasi-experiment with undergraduate Computer Science students compared the catalog with guidelines published by practitioners and investigated its effects on perceived learning. Finally, a Multivocal Literature Review identified additional anti-patterns in informal sources, including blogs, technical websites, and online communities, resulting in a more comprehensive and practice-informed version of the catalog.

3 Results

After collecting feedback from participants and incorporating new anti-patterns from grey literature, we evolved our catalog and created its final version with 27 anti-patterns. Drawing on the collaborative repository model proposed by Bogner et al. [1], we developed an web application to showcase all anti-patterns.¹

4 Contributions

This research resulted in the following publications:

- **Published papers:**

¹<https://mfe-anti-patterns.online/micro-frontends-anti-patterns/#/catalog>

- (1) *A Catalog of Micro Frontends Anti-patterns* [12]: Catalog's proposal and evaluation by practitioners through a Personal Opinion Survey. Published at the IEEE/ACM International Conference on Software Engineering (ICSE) 2025 – Research Track.
- (2) *Evaluating Strategies for Teaching Micro Frontends: Do Anti-patterns Help?* [13]: Experience report on teaching Micro Frontends to undergraduate Computer Science students, along with a controlled experiment comparing the catalog to guidelines provided by developers on the internet. Published at the XXXIX Brazilian Symposium on Software Engineering (SBES) 2025 – Education Track, and received a Distinguished Paper Award.
- (3) *Teaching Micro Frontends: Insights from Two Controlled Experiments on Guidelines and Anti-patterns* [14]: Extended version of the paper published in SBES Education Track [13] due to the invitation after receiving the Distinguished Paper Award. We conducted a second controlled experiment to compare a new version of the catalog with the original one. Published at the Journal of Software Engineering Research and Development (JSERD).
- (4) *Designing and Evaluating a Catalog of Micro Frontends Anti-patterns* [11]: Extended abstract summarizing the thesis after its selection among the ten best Master's theses in Computing defended in Brazil in 2025 by the Brazilian Computer Society (SBC). Published at the SBC's Concurso de Teses e Dissertações (CTD-SBC).

• Submitted papers:

- (1) *What Do We Know About Micro Frontends Anti-patterns? A Multivocal Review of Practitioner Knowledge*: Report of the Multivocal Literature Review we conducted to expand the catalog by adding anti-patterns proposed by practitioners in Grey Literature sources. We submitted this paper to the Information and Software Technology (IST) journal.

5 Conclusion

By proposing and evaluating an MFE anti-pattern catalog, this thesis advances Software Engineering by turning fragmented practitioner experience around an emerging architectural style into structured, reusable, and empirically grounded knowledge. The catalog formalizes recurring failure modes by clearly defining problems, symptoms, consequences, and mitigation strategies, creating a shared conceptual basis for analyzing, comparing, and evolving MFE. Beyond documentation, this work provides empirical evidence of how these problems manifest in real systems and how practitioners use the catalog in day-to-day work, thereby strengthening the scientific understanding of MFE-specific constraints and informing future research on measurement, tool support, and automated detection. In doing so, the thesis contributes both to theory, by characterizing phenomena and relationships in MFE architectures, and to practice, by enabling knowledge transfer that can improve architectural decision-making, education, and the quality of MFE-based systems at scale.

For future work, we plan to conduct a participatory case study to analyze how practitioners use the catalog and to provide empirical evidence of its impact on the overall quality of MFE architectures.

We also aim to examine the effects of the proposed solutions, both positive and negative, investigating whether their application may introduce new anti-patterns. Regarding anti-pattern detection, we intend to formalize the definition of anti-patterns using a formal notation or specification language, enabling precise representation and systematic analysis. We plan to develop automated detection tools to help identify anti-patterns efficiently. Finally, we aim to explore how the catalog can serve as a foundational knowledge base for building AI-powered agents that assist practitioners in making informed architectural decisions in MFE.

Artifact Availability

The artifacts produced by this thesis are available at each paper's supplementary material: A Catalog of Micro Frontends Anti-patterns², Evaluating Strategies for Teaching Micro Frontends: Do Anti-patterns Help?³, and Teaching Micro Frontends: Insights from Two Controlled Experiments on Guidelines and Anti-patterns⁴.

Acknowledgments

This paper was supported by CAPES-PROEX, Finance Code 001; FA-PEAM through project POSGRAD 2026/2027; and CNPq processes 314797/2023-8, 443934/2023-1, and 445029/2024-2.

References

- [1] Justus Bogner, Tobias Boeck, Matthias Popp, Dennis Tschachlov, Stefan Wagner, and Alfred Zimmermann. 2019. Towards a collaborative repository for the documentation of service-based antipatterns and bad smells. In *2019 IEEE International Conference on Software Architecture Companion (ICSA-C)*. IEEE, 95–101.
- [2] Namiot Dmitry and Sneps-Snepp Manfred. 2014. On micro-services architecture. *International Journal of Open Information Technologies* 2, 9 (2014), 24–27.
- [3] Nicola Dragoni, Saverio Giallorenzo, Alberto Lluch Lafuente, Manuel Mazzara, Fabrizio Montesi, Ruslan Mustafin, and Larisa Safina. 2017. Microservices: yesterday, today, and tomorrow. *Present and ulterior software engineering* (2017), 195–216.
- [4] Thomas Erl. 2016. *Service-oriented architecture: analysis and design for services and microservices*. Prentice Hall Press.
- [5] Michael Geers. 2020. *Micro Frontends in Action*. Simon and Schuster.
- [6] Rick Kazman, Yuanfang Cai, Michael W Godfrey, Cesare Pautasso, and Anna Liu. 2023. A Better Way to Teach Software Architecture. In *Software Architecture: Research Roadmaps from the Community*. Springer, 101–110.
- [7] James Lewis and Martin Fowler. 2014. Microservices a definition of this new architectural term. <https://www.martinfowler.com/articles/microservices.html>
- [8] Sômulô Nogueira Mafra, Rafael Ferreira Barcelos, and Guilherme Horta Travassos. 2006. Aplicando uma metodologia baseada em evidência na definição de novas tecnologias de software. In *Anais do XX Simpósio Brasileiro de Engenharia de Software*. SBC, 239–254.
- [9] Luca Mezzalana. 2021. *Building Micro-Frontends*. O'Reilly Media, Inc.
- [10] Severi Peltonen, Luca Mezzalana, and Davide Taibi. 2021. Motivations, benefits, and issues for adopting micro-frontends: a multivocal literature review. *Information and Software Technology* 136 (2021), 106571.
- [11] Nabson Silva and Tayana Conte. 2026. Designing and Evaluating a Catalog of Micro Frontends Anti-patterns. In *Concurso de Teses e Dissertações da SBC (CTD-SBC)*. SBC, 60–69.
- [12] Nabson Silva, Eriky Rodrigues, and Tayana Conte. 2025. A Catalog of Micro Frontends Anti-patterns. In *2025 IEEE/ACM 47th International Conference on Software Engineering (ICSE)*. 616–616. doi:10.1109/ICSE55347.2025.00079
- [13] Nabson Silva, Eriky Rodrigues, and Tayana Conte. 2025. Evaluating Strategies for Teaching Micro Frontends: Do Anti-patterns Help?. In *Simpósio Brasileiro de Engenharia de Software (SBES)*. SBC, 522–532.
- [14] Nabson Silva, Eriky Rodrigues, and Tayana Conte. 2026. Teaching Micro Frontends: Insights from Two Controlled Experiments on Guidelines and Anti-patterns. In *Journal of Software Engineering Research and Development*. SBC.
- [15] Marco Tulio Valente. 2020. Engenharia de software moderna. *Princípios e Práticas para Desenvolvimento de Software com Produtividade* 1, 24 (2020).

²<https://github.com/nabsonp/ICSE-2025>

³<https://zenodo.org/records/16394584>

⁴<https://figshare.com/s/7fd92ace78cbb80a024b>