

# 1 Monolithic Architecture

Before going into Microservices and the Microservice architecture, the Monolithic architecture approach must be explained first. The Monolithic architecture approach was till recently the preferred design option for software. In a Monolithic application all different components and functions are combined into one indivisible program[1]. While individual components might be developed separately, they remain tightly coupled[2] and any change completed in any of them requires the whole program to be rebuild and redeployed[3]. More often than not development in one component requires functional changes in multiple other, adding on the development cost, complicating the build and testing process and inducing delays in deployment, while a bug in any one component can potentially halt the entire application's operation. Additionally Monolithic applications usually have large codebases, which can be cumbersome when implementing changes and difficult to manage over time[2]. Another major issue with Monolithic applications is scalability. Usually different components have conflicting resource requirements but because of the unified design all requirements must be handled together making scaling up the application almost impossible. Despite the many drawbacks of Monolithic architecture there ...

## References

- [1] Konrad Gos and Wojciech Zabierowski. The comparison of microservice and monolithic architecture. pages 150–153, 04 2020.
- [2] Ivy Wigmore Rahul Awati. What is monolithic architecture? <https://www.techtarget.com/whatis/definition/monolithic-architecture>, 2022.
- [3] Freddy Tapia, Miguel Ángel Mora, Walter Fuertes, Hernán Aules, Edwin Flores, and Theofilos Toulkeridis. From monolithic systems to microservices: A comparative study of performance. *Applied Sciences*, 10(17), 2020.