

# **Comp 680 Seminar Proposal**

Team 3

Edgar Lopez-Garcia

Ashot Chobanyan

[https://elopezga.github.io/comp\\_680\\_website/](https://elopezga.github.io/comp_680_website/)

## Overview

### Microservices and Software Development Life Cycle

Microservices is an architectural pattern that aims at breaking up application services into individually operational services. Microservices allow more easier manageable development and maintenance of software. As opposed to a monolithic architectural style, microservices makes each service within an application its own independent piece of software. For instance, in a social media application, the search functionality and news feed can be made into separate services that are decoupled from each other. The decoupling aspect of microservices allows for highly flexible and reliable systems. For our seminar, we will discuss the microservices architecture, its comparison with a monolithic architecture, the tools and techniques used in developing microservices, and case studies of microservices architecture in real-world applications. Refer to the list below for a complete list of topics that will be covered by the seminar.

### Seminar 1 Topics

- Monolithic architecture<sup>[2]</sup>
- Microservices architecture<sup>[1]</sup>
- Benefits of microservices<sup>[4]</sup>
- SDLC implications (Monolithic & Microservices)<sup>[12]</sup>
- Tools, Platforms, Frameworks, and Services<sup>[13][14]</sup>

### Seminar 2 Topics

- Microservices in-depth<sup>[17]</sup>
- Types of microservices<sup>[14]</sup>
- Implementation techniques<sup>[13][15][16][18]</sup>
- Case studies<sup>[11]</sup>

## References

[1] <https://microservices.io/>

[2] <https://microservices.io/patterns/monolithic.html>

[3]

<https://medium.com/koderlabs/introduction-to-monolithic-architecture-and-microservices-architecture-b211a5955c63>

[4]

<https://medium.freecodecamp.org/monolith-vs-microservices-which-architecture-is-right-for-your-team-bb840319d531><https://medium.freecodecamp.org/monolith-vs-microservices-which-architecture-is-right-for-your-team-bb840319d531>

- [5] [http://www.codingthearchitecture.com/2014/11/19/what\\_is\\_a\\_monolith.html](http://www.codingthearchitecture.com/2014/11/19/what_is_a_monolith.html)
- [6] <https://articles.microservices.com/monolithic-vs-microservices-architecture-5c4848858f59>
- [7] <https://ieeexplore.ieee.org/abstract/document/7030212>
- [8] <https://www.thorntech.com/2017/12/microservices-vs-monoliths-whats-right-architecture-software/>
- [9] <https://dzone.com/articles/5-steps-to-successfully-prepare-for-microservices>
- [10] <https://www.slideshare.net/TriNimbus/chris-munns-devops-amazon-microservices-2-pizza-teams-50-million-deploys-a-year>
- [11] <https://www.nginx.com/blog/adopting-microservices-at-netflix-lessons-for-team-and-process-design/>
- [12] <https://blog.armory.io/how-microservices-help-continuous-delivery-in-the-sdlc-part-one/>
- [13] <https://www.devbridge.com/articles/a-6-point-plan-for-implementing-a-scalable-microservices-architecture/>
- [14] <https://aws.amazon.com/microservices/>
- [15] <https://techbeacon.com/app-dev-testing/8-best-practices-microservices-app-security>
- [16] <https://www.apiacademy.co/lessons/2016/06/api-design-304-api-design-for-microservices>
- [17] <https://dzone.com/articles/patterns-for-microservices-sync-vs-async>
- [18] <https://cdelmas.github.io/2015/11/01/A-comparison-of-Microservices-Frameworks.html>