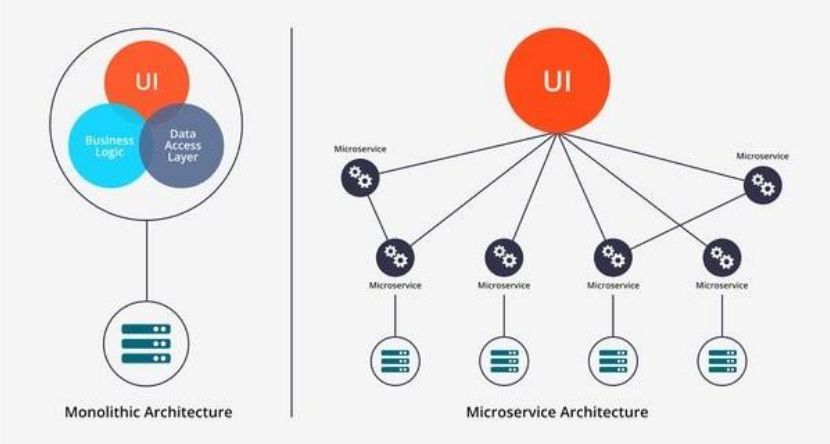
# ECE444-PRA4: Architecture Design.






## Learning goal:

Exploring alternatives of the architecture of your project 1. Students need to explore materials and finish the 3 activities.

1. **Activity 1: (Microservices)** Decompose your existing prototype into at least three (3) services that can be deployed separately. Decide on a reasonable decomposition and use remote procedure calls (e.g., REST API) to communicate between services.
2. **Activity 2: (Monolithic**) Redesign your system in a Monolithic way, showing all the components inside the monolithic architecture.
3. **Activity 3:** Compare the trade-off between alternatives (microservices vs monolithic), in terms of ONE (1) ***quality*** **requirement** that you picked in your Milestone 2 report.

In both diagrams and text, document the alternative architecture of your system. Create diagrams that show ONE (1) view with which it is possible to reason about the quality requirements you considered in your milestone 2.

Compare the trade-off between alternatives, in terms of ONE (1) ***quality*** **requirement**. The architecture report should include meaningful legends for each diagram included in the report.

**There is a strict page limit of 3 pages (excluding references), single-sided, 12pt, double-line space.**

# Frequently Asked Questions

1. Any tool recommendations for drawing architectural design diagram
    - You are free to use any tool for drawing design diagrams.
    - Here are some of our favorites
        - LucidCharts (https://www.lucidchart.com/pages/)
        - Diagrams.net (https://app.diagrams.net/)
        - SmartDraw (https://smartdraw.com/)
        - Draw.io (https://draw.io/)
2. Do the monolithic and micro-services architectures need to have the same components?
    - You can have the same or different components and tools for each architecture, but they should cover the same functional and non-functional requirements listed in your Milestone 2: requirement engineering.

# Evaluation

This is an open-ended assignment. As long as your architecture document is easy to read without ambiguity, it is a good document.

- A precise and complete description of your proposed architecture (diagrams and text). Clarity is paramount. Choose a consistent level of abstraction to document; do not provide extensive detail on one part of the architecture but gloss over important high-level design decisions on another.
- An explicit comparison between your architecture and an alternative design. The comparison should be supported with technical arguments.
- A discussion that avoids handwavy, incomplete, or non-sequitur arguments: “Approach X is more reliable because there are fewer messages sent.” (How do few messages translate to higher reliability? More reliable compared to what: the original architecture or approach Y?)
- Grading will be based on the completion and correctness of the architectures, additionally, the TA may ask questions to assess your understanding and design choices.
- Provide references if needed.