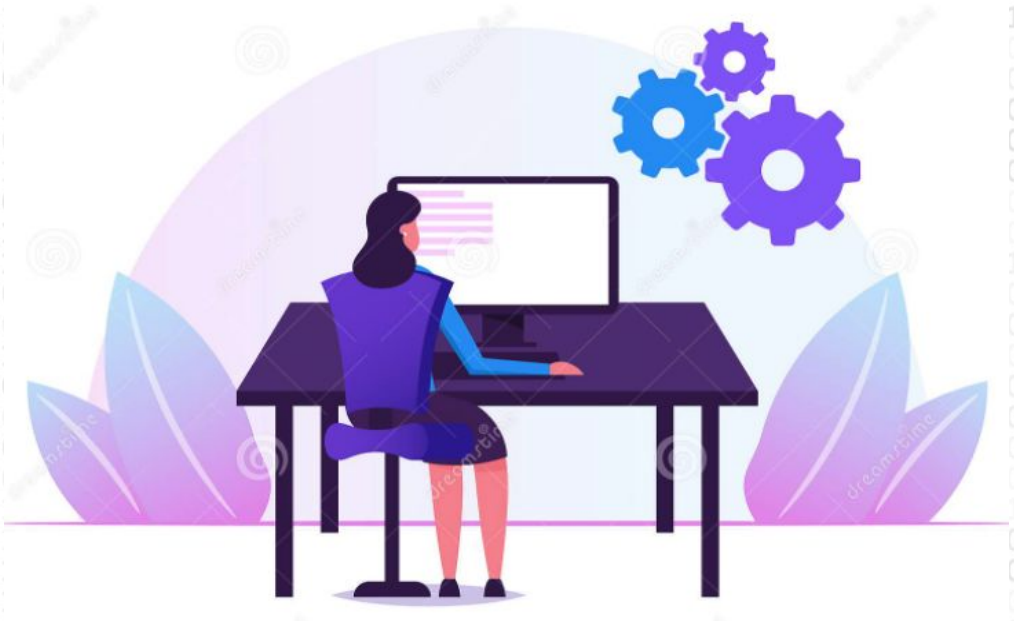
Welcome!



- We'll start in a moment :)
- We are NOT recording tonight's event. We may plan to take screenshots for social media.
- ***If you want to remain anonymous***, change your name & keep video off.
- We'll introduce the hosts & might break in-between for Q/A.
- We will make some time for Q&A at the end of the presentation as well.
- You can come prepared with questions.
- Feel free to take notes.
- Online event best practices:
  - Don't multitask. Distractions reduce your ability to remember concepts.
  - Mute yourself when you aren't talking.
  - We want the session to be interactive.
  - Feel free to unmute & ask questions.
- Turn on your video if you feel comfortable.

- *Disclaimer: Speaker doesn't knows everything!*

WWCode Digital + **Backend**

Backend Study Group

July 28, 2022

WOMEN WHO CODE

Introduction & Agenda

- Welcome from WWCode!
- Our mission: Inspiring women to excel in technology careers.
- Our vision: A world where women are representative as technical executives, founders,VCs, board members and software engineers.

Prachi Shah
Speaker,
Senior Software Engineer, Metromile
Director, WWCode SF

Harini Rajendran
Co-host,
Software Engineer, Confluent
Lead, WWCode SF

- Microservice vs. Monolith architecture:
- What is a Monolith architecture?
- What is a Microservice architecture?
- Comparison of the two architectures.
- Q & A and open discussion.

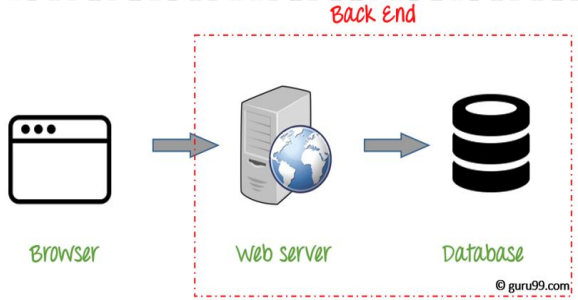
Disclaimer:
- Sessions can be heavy!
- Lots of acronyms.
- Speaker doesn't know everything.

WOMEN WHO CODE

Backend Engineering

- Design, build and maintain server-side web applications.

- Common terms: Client-server architecture, networking, APIs, web frameworks, platform, micro-service, databases, web fundamentals, operating systems, etc.



- Tech Stack: Java, PHP, .NET, C#, Ruby, Python, REST, AWS, Node, SQL, NoSQL.

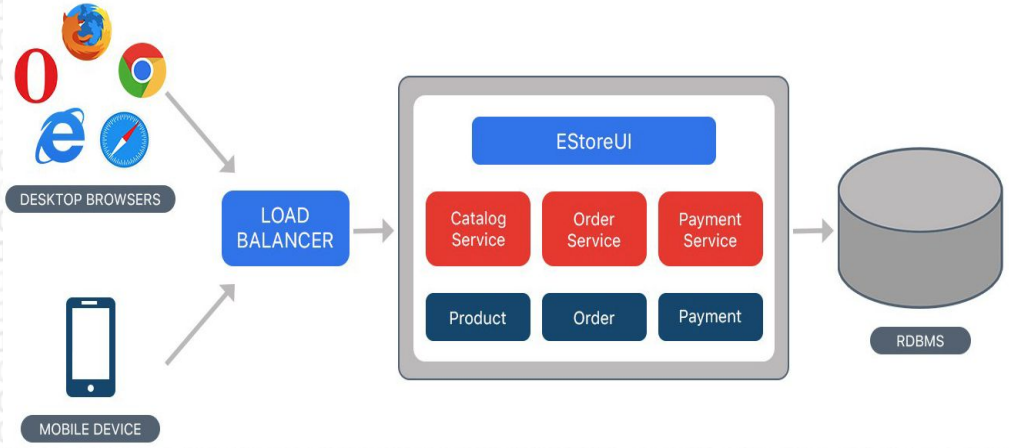
- Other domains: Front end engineering, full stack engineering, design & user experience, mobile development, devOps engineering, machine learning, etc.

- Examples: Amazon Online Shopping, Instagram, Weather app, etc.

WOMEN WHO CODE

Monolith Architecture

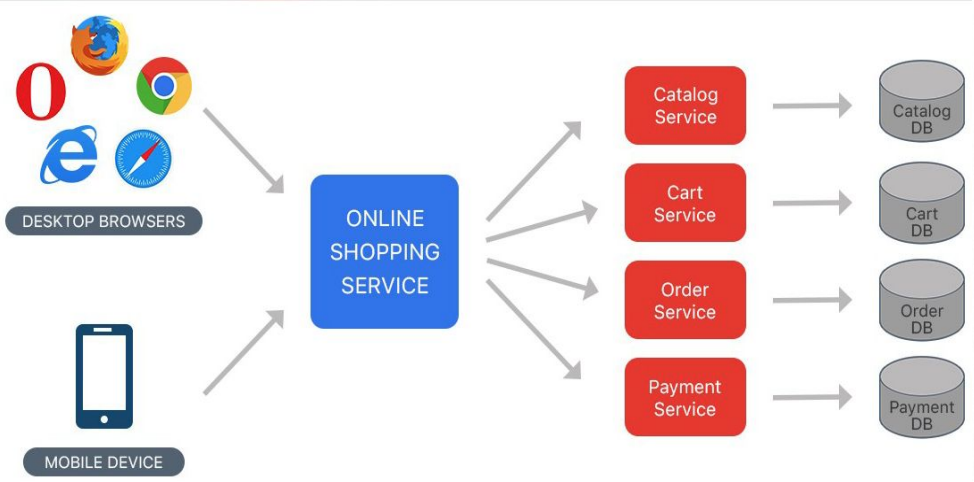
- “All in one place”.
- Components and functions are tightly coupled.
- Multiple features, programs tasks in on codebase
  large codebase.
- Support for multiple platforms and languages.
- Consists of APIs, authentication,
  database management, business logic,
  application configuration, data presentation.
- Pros:
  - Best for starter projects that evolve over time.
  - Easy to test, debug and configure.
- Cons:
  - Maintenance is difficult as the code grows.
  - Testing the entire application when fixing bugs.
  - Re-deploy entire application on any new changes.
  - Hard to support new and upcoming technologies.
- Example: E-commerce application monolith does it all.



WOMEN WHO CODE

Microservice Architecture

- Modular programming.
- Smaller applications deployed independently.
- Each application server one major functionality.
- Components and functions are loosely coupled.
- Multiple features, programs tasks in different smaller codebases.
- Support for multiple platforms and languages.
- Services for REST APIs, authentication, database management, business logic, application configuration, data presentation.
- Pros:
  - Easy to maintain and deploy.
  - Easy to scale vertically.
  - Supports CI/CD (Continuous integration/ Continuous delivery).
- Cons:
  - Understand and configure a distributed system.
  - Deployment and monitoring complexity.
- Example: E-commerce application microservice does many different things.



WOMEN WHO CODE

Monolith vs. Microservice

Monolith:

- Tightly coupled functionalities.
- Easy to deploy and test (initially).
- Hard to maintain, slower bug fixes.
- Easy to deploy as one application.
- Does not scale to support new technologies. Less reliance on vendors.
- Slower code to production.
- Good as a starter application.
- Spaghetti code syndrome.
- More ramp-up time for engineers.
- Hard to standardize software processes.

Microservice:

- Loosely coupled functionalities.
- Easy to deploy and test (service grows).
- Easy to maintain, faster bug fixes.
- Deployment and monitoring is complex.
- Easy to support new technologies. More reliance on vendors.
- Faster code to production.
- Suitable for complicated systems.
- Modular programming.
- Faster code delivery.
- Standardization of software development.

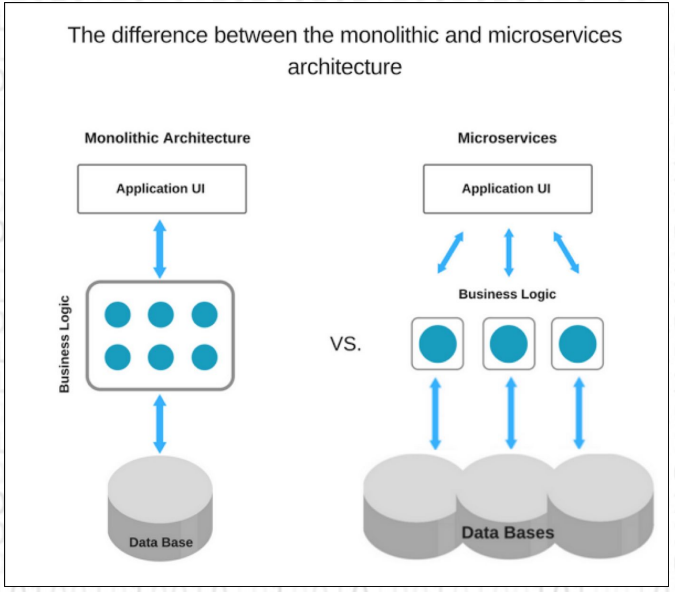
WOMEN WHO CODE

Code Demo

Java Microservice with a Database and CRUD operations

WOMEN WHO CODE

Monolith & Microservice



Key Takeaways:

- Monolith best for starter application with minimum functionalities and small applications which do not evolve much.
- Microservices best to develop complicated functionalities, that evolve over time.
- Microservices suitable for applications that deal with lots of data.
- Microservices suitable for improved application monitoring and CI/CD.
- Microservices provide faster ramp-up time and quicker code delivery to production.
- Most modern applications are Microservices and MVC-architecture based.

WOMEN WHO CODE

Backend Engineering

Resources:

- Introduction to Monolithic Architecture and MicroServices Architecture

Backend Study Group:

- Presentations on GitHub and session recordings are found on WWCode YouTube channel
- Upcoming session:
    - August 16th, 2022 about OOP & Design Principles (using Java)
    - August 25th, 2022 about System Design mock interview
- Technical Tracks and Digital Events.
- Get updates – join the Digital mailing list!
- Have questions?
    - Contacts us at: contact@womenwhocode.com
    - Join our Slack workspace and join *#backend-study-group*!

You can unmute and talk or use the chat.

WOMEN WHO CODE