

Lecture 3 - 19th Sep 2025

/// Monolithic Systems.

what is it?

- One .zip deployed as whole - with all (models, views, controllers).
- Shared memory & database
- Advan: - Easy to start development
- one file = one Deploy.
- Disadvan: One bug → All crash

Deployment is Bottleneck



instead of this, use

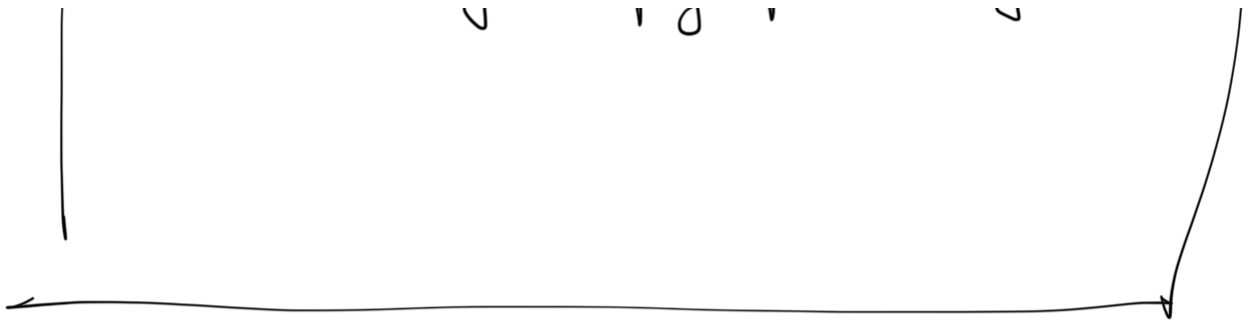
Micro-Services

- small INDEPENDENT services → each responsible for one specific business function
- they communicate with each other using APIs (HTTP, REST, gRPC).
- smaller apps instead of one big app.
- each has separate DB
- each can be deployed and scaled independently.



Microservices ++ : Serverless Components

- no servers to manage. yourself.
- auto scaling → pay-per-usage



Swagger.

- tool for documentation of APIs

AWS : API Gateway

- FRONT-DOOR for our backend with Security checks & job of "Receptionist".
- ROUTES requests to CORRECT Lambdas

~~///~~ AWS : Lambda

→ Serverless : bcoz we dont need to create & manage a server.

~~///~~ Connecting

→ API Gateway → edit integration →
choose "Lambda func".

→ Enable "lambda proxy integration".

→ Choose "lambda function".

→ Submit & Done



We can test the "api + lambda" working.



But now we need to Deploy the API.

but BEFORE that ?

① click "Enable CORS".

② click "Deploy API".

— this will give us an "invoke URL"
which can be used to access the
working

— we can also create APIs directly from
API Gateway using "Create Resource".



Once everything is done → click on

"Generate SDK"

↪ when we get the SDK → we can directly link it to the frontend.



Replace the SDK in the code by the one downloaded.

01:03:00 -

RECAP.

If one lambda sends output to other lambda → It can act as a BOTTLENECK

↓
due to law of "SLOWEST Component".



We need to DECOUPLE them - by putting in an SQS in between.

--

// Dremel Paper Review.

- ① Storing data in columnar storage.

// PDF Content

= SDK (Software Dev. Kit) -

- collecⁿ of tools, libraries and code samples → that helps devs interact with platform/service/API easily.

= Async

- Client & Server **DONT NEED** to communicate **at ONCE**.
- Process :
 - client sends requests
 - forgets about it,

- Server does processing .

- Informs client LATER using
callback, event, webhook, message queues

↓
AWS SQS