Serverless
– are you ready for the revolution?

Michał Jankowski

about
me

Michał Jankowski

Objectivity
Bespoke Software Specialists

architect / software developer / team leader

traveller / photographer

www.jankowskimichal.pl

mail@jankowskimichal.pl

@JankowskiMichal

github.com/MichalJankowskii

aim.

Exchange our knowledge regarding Serverless approach and try to think about how it can change our way of designing software.

presentation
agenda

serverless

A short introduction to this approach.

revolution

Let’s think how serverless can change the way how we build software.

presentation
agenda

in azure

A brief introduction to serverless components that are available in Azure cloud.

real life case

A short journey during which we will start from prototype and we will try to achieve production ready state.

do you have any
experience with serverless?

serverless.

types of approaches

01 Do I really know how hardware works? What hardware specification should be delivered?

02 Am I a good system administrator? When I should update servers' OS?

03 Finally I can focus on features development.

On-Premises	IaaS	PaaS	Serverless
Applications	Applications	Applications	Applications
Data	Data	Data	Data
Runtime	Runtime	Runtime	Runtime
Middleware	Middleware	Middleware	Middleware
O/S	O/S	O/S	O/S
Virtualization	Virtualization	Virtualization	Virtualization
Servers	Servers	Servers	Servers
Storage	Storage	Storage	Storage
Networking	Networking	Networking	Networking

Managed by us

Managed by vendor

approaches summary

	IAAS	PAAS	CONTAINER	SERVERLESS
Scale Refers to the unit that is used to scale the application	VM	Instance	App	Function
Abstract Refers to the layer that is abstracted by the implementation	Hardware	Platform	OS Host	Runtime
Unit Refers to the scope of what is deployed	VM	Project	Image	Code
Lifetime Refers to typical runtime of a specific instance	Months	Days to months	Minutes to days	Milliseconds to Minutes
Responsibility Refers to the overhead to build, deploy, and maintain the application	Applications, dependencies, runtime, and operating system	Applications and dependencies	Applications, dependencies, and runtime	Function

serverless characteristics

server abstraction

There is no server managing tasks.

event driven

Function does not work when there is no event triggering it. It can also instantly scale up.

microbilling

Pay only when there are events. But think about DDOS on your wallet.

productivity

Reduce tasks related to infrastructure. You can focus on development activities.

focus on features

And then you are able to focus on business logic of your app.

faster time to market

All items mentioned together allow you to reduce time to market.

time of starting

We will need additional time for our function start. Normal application are always ready for response.

think about state

Functions are stateless. You should save somewhere state if you need.

local environment

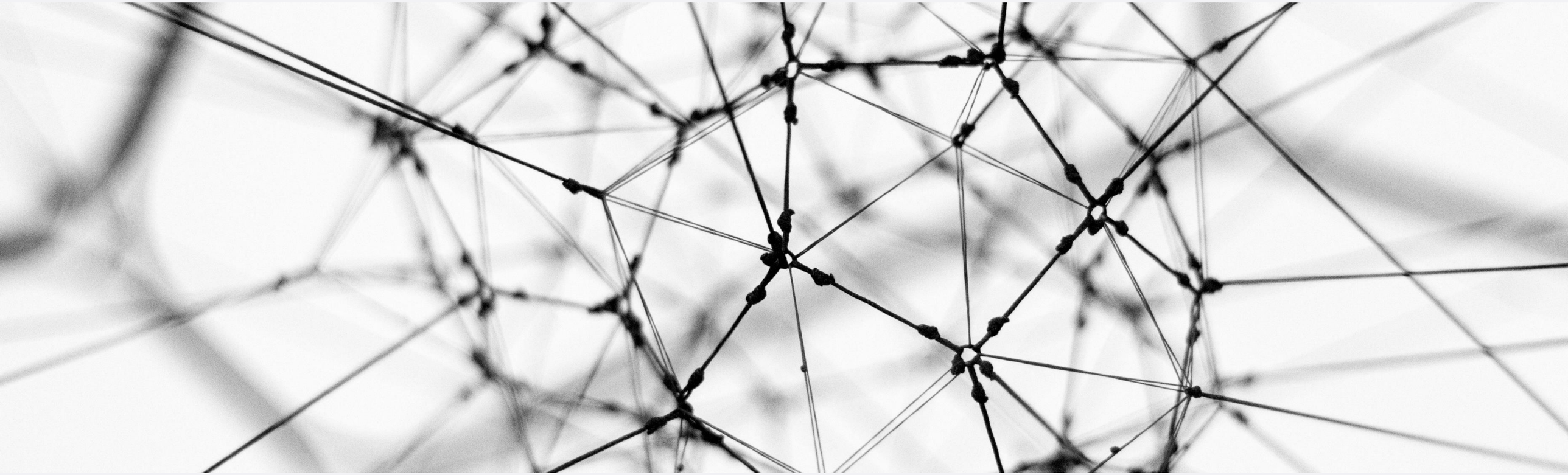
It is not so easy to start your function locally and it can be run only under Windows.

vendor locking

It will be hard to change your vendor in latter stage of your application life.

revolution.

what are your biggest challenges?



do we design it correctly

“Many of the solutions that we consider best practice are solutions for problems that no longer apply”

Gojko Adzic, MindMup / Claudia.js

do we design it **correctly**

01 architecture optimised for reserved resources vs bundling into apps

02 paid for reserved resources vs utilised capacity

03 optimized for quick fail-over vs time to start

04 difficult to replicate "production" vs multiple version of functions

05 layered architecture vs smart composition

06 high vs low cost

Play arbitrage with different charging models

are we ready
for the revolution?

LEGAL: SERVICE LEVEL AGREEMENTS > Functions

SLA for Functions

Last updated: November 2016

For Function Apps running on App Service Plans we guarantee that the associated Functions compute will be available 99.95% of the time.

No SLA is provided for Functions Apps running under Consumption Plans.

in Azure.

serverless
in Azure

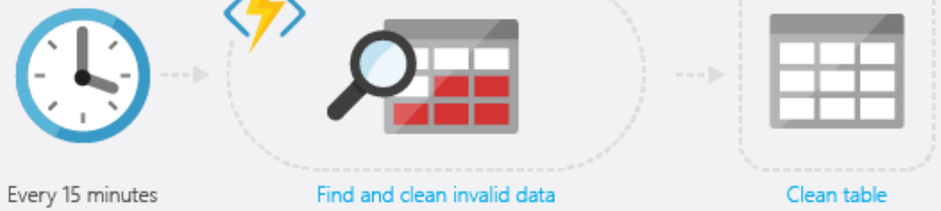
Azure Functions

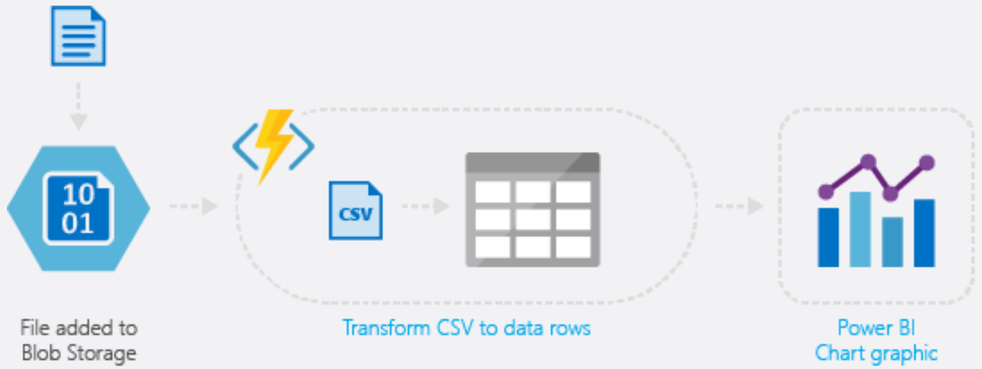
An event-based serverless compute experience to accelerate your development. Scale based on demand and pay only for the resources you consume.

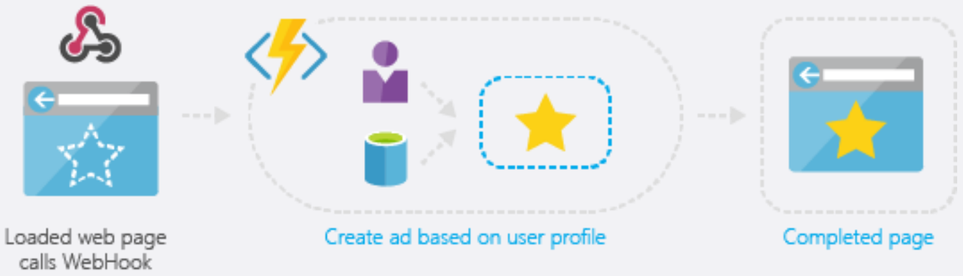
Event Grid

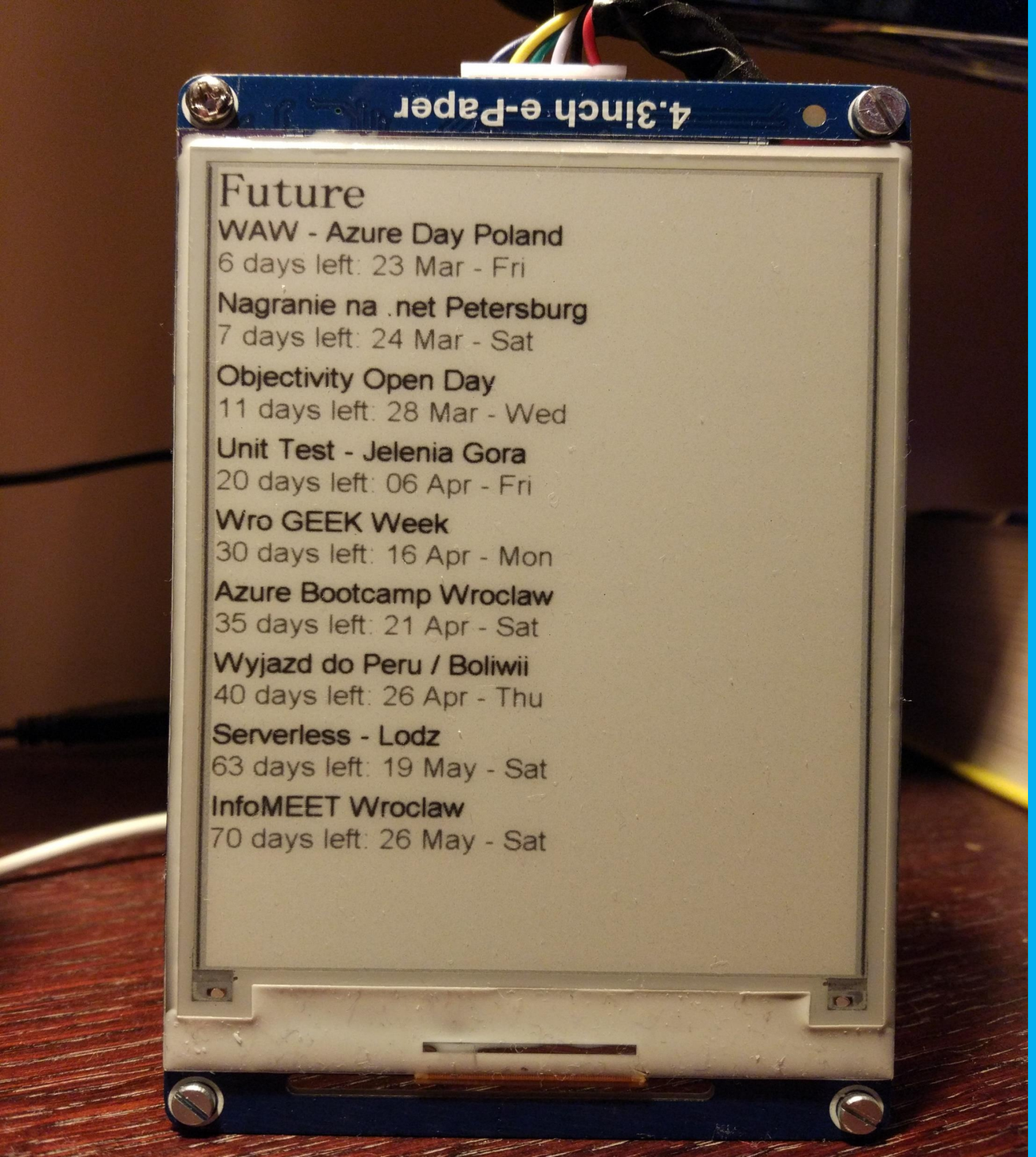
A single service for managing routing of all events from any source to any destination. Designed for high availability, consistent performance and dynamic scale. Event Grid lets you focus on your app logic rather than infrastructure.

common scenarios





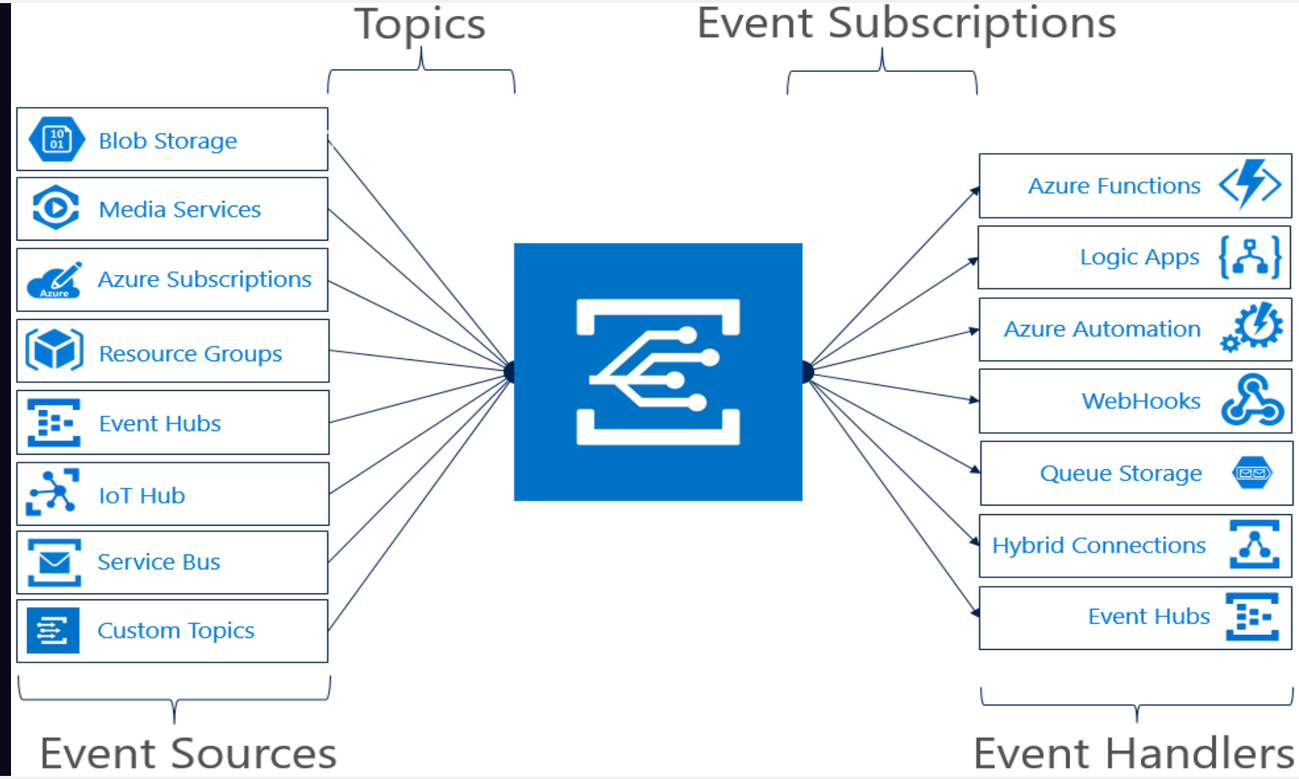




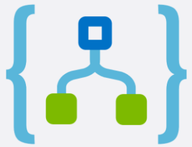
- Azure Function as a glue
- My favourite usage scenario
- Arduino
  → Azure Function
  → ToDoist

event grid

- a single service for managing routing of all events from any source to any destination
- advanced filtering - filter on event type or event publish path to ensure event handlers only receive relevant events
- reliability – utilize 24-hour retry with exponential backoff to ensure events are delivered and ensure that message will be delivered once and only once
- high throughput – build high-volume workloads on Event Grid with support for 10 millions of events per second
- supports only subset of apps
- pay-per-event - pay only for the amount you use Event Grid



serverless
in Azure



Logic Apps

Provide a way to simplify and implement scalable integrations and workflows in the cloud. It provides a visual designer to model and automate your process as a series of steps known as a workflow.

Flow

Is a service that allows you to create automated workflows between your favourite applications and services to synchronize files, get notifications, collect data, and more.

Cosmos DB

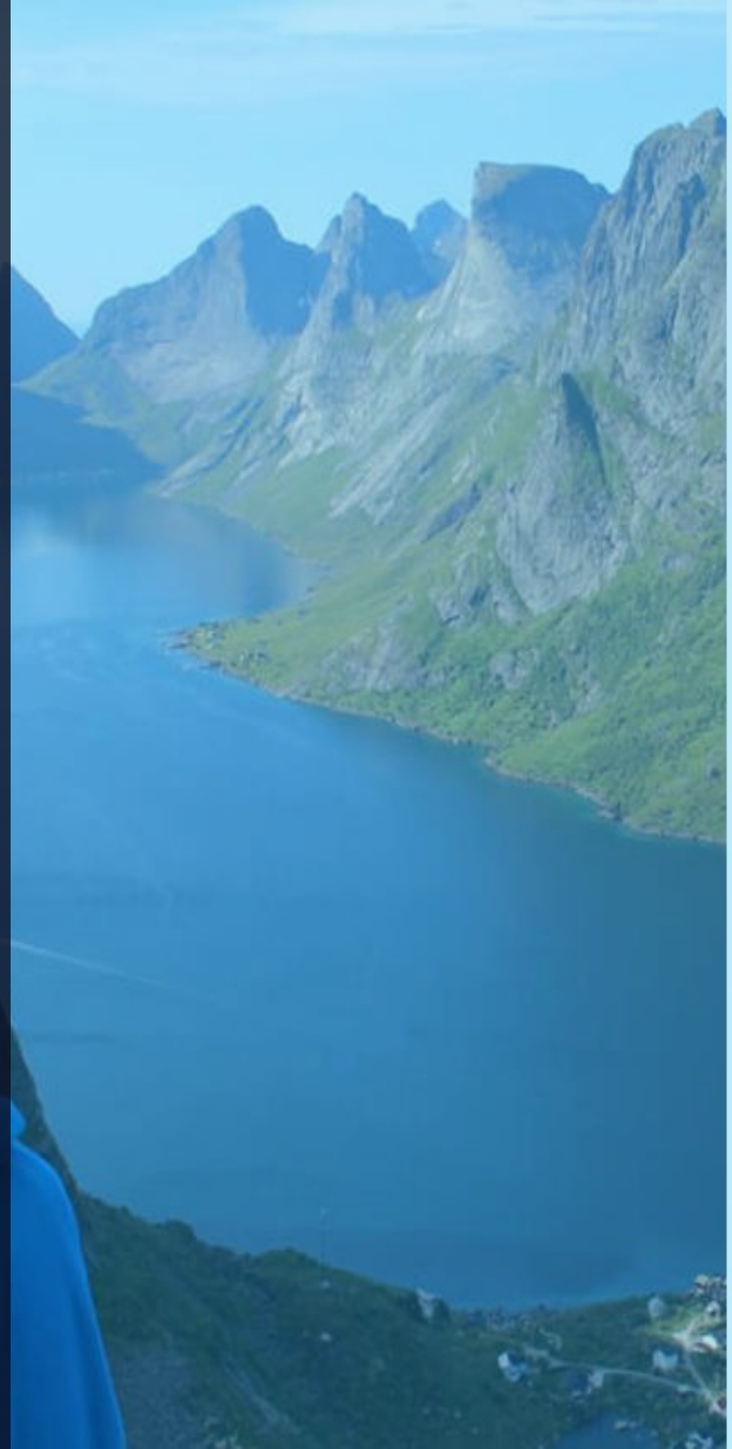
Was built from the ground up with global distribution and horizontal scale at its core. It offers turnkey global distribution with multi-master support across any number of Azure regions by transparently scaling and replicating your data wherever your users are.

real case.

case.

We will provide Azure Function backend for registration form that will:

- validate data
- store customer data
- send thank you message
- send SMS to us each time new customer will register



Get The Best Ticket Today!

You can add unlimited fields directly from HTML

Your name

Your surname

Your county

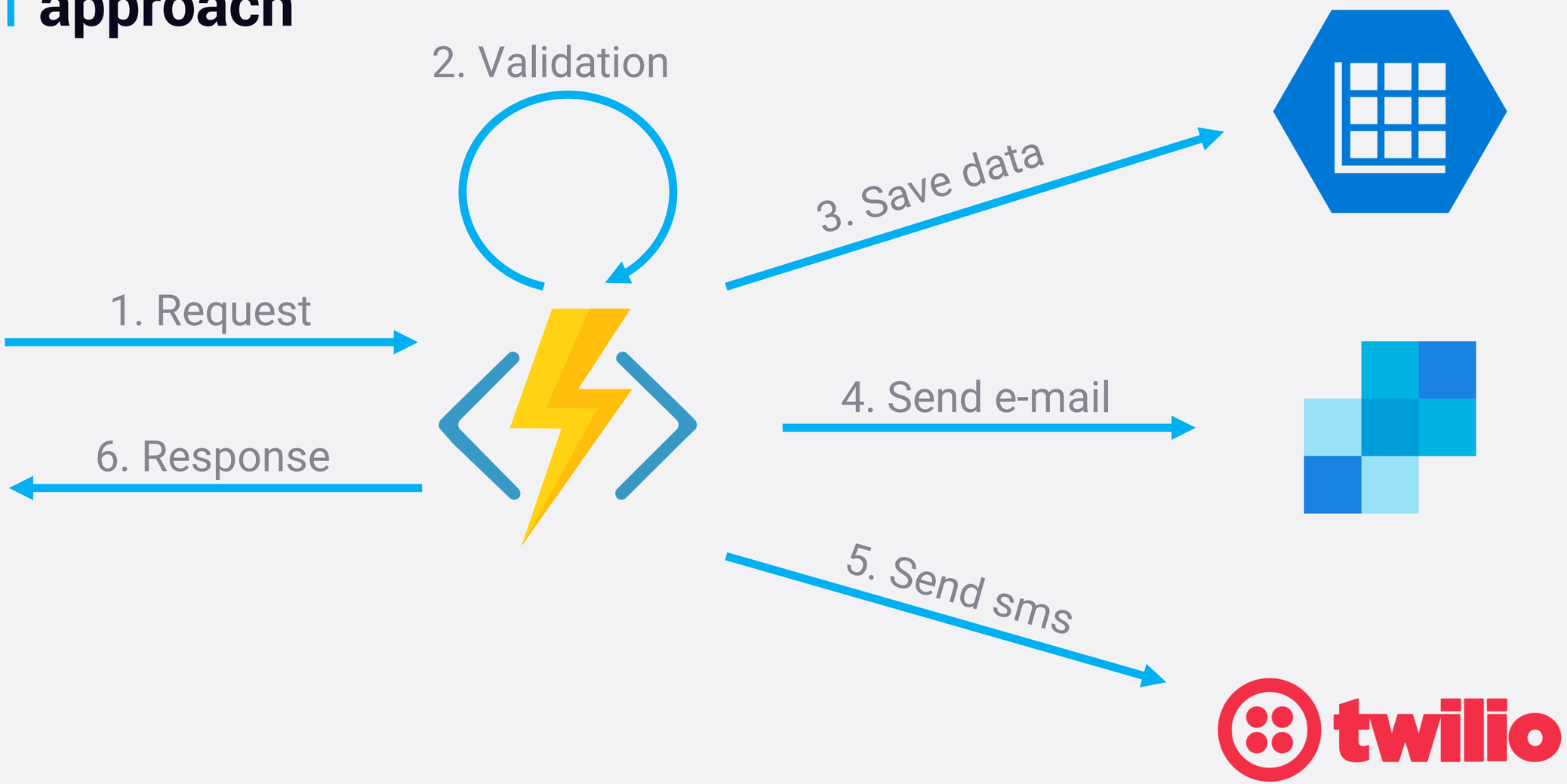
Your e-mail

Your birth year

Send Information

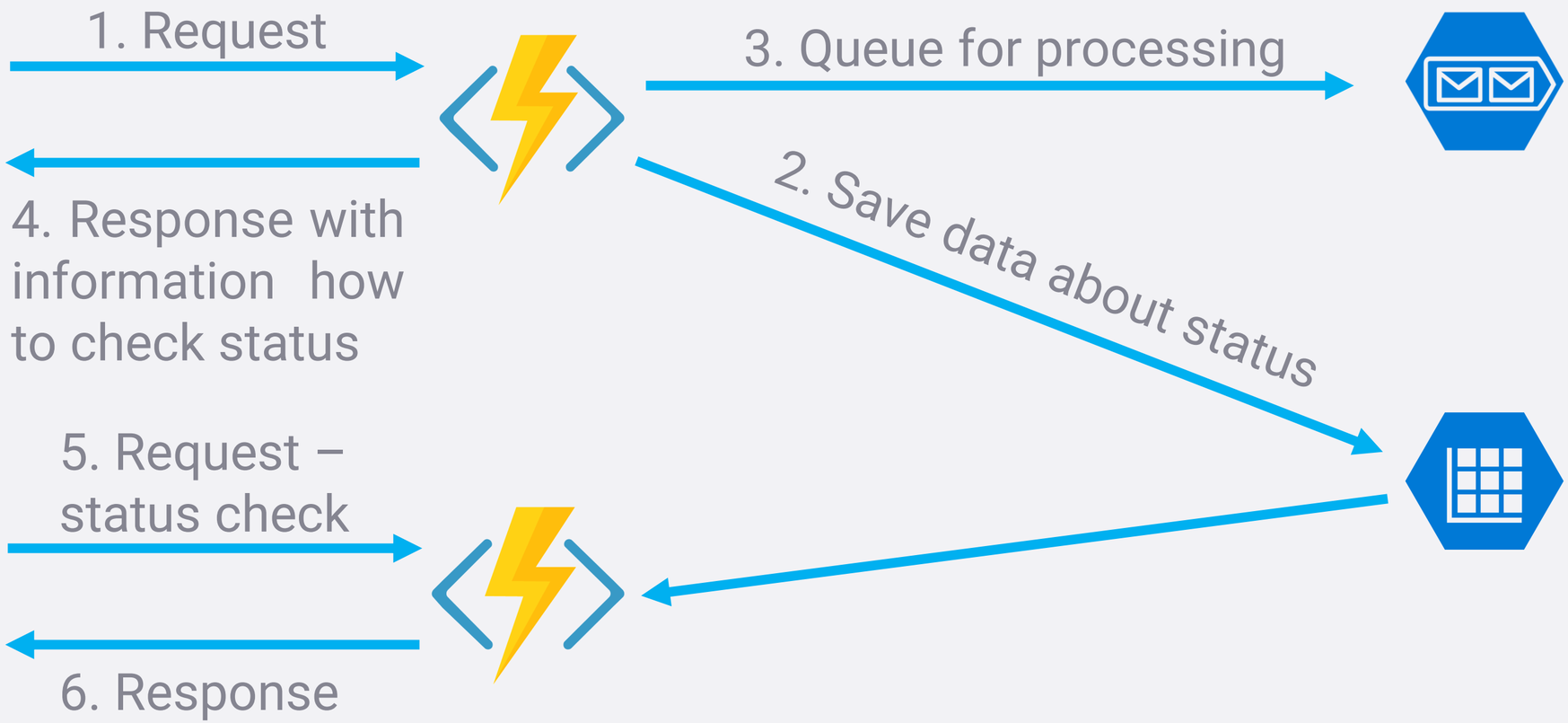
prototype
solution I

first approach

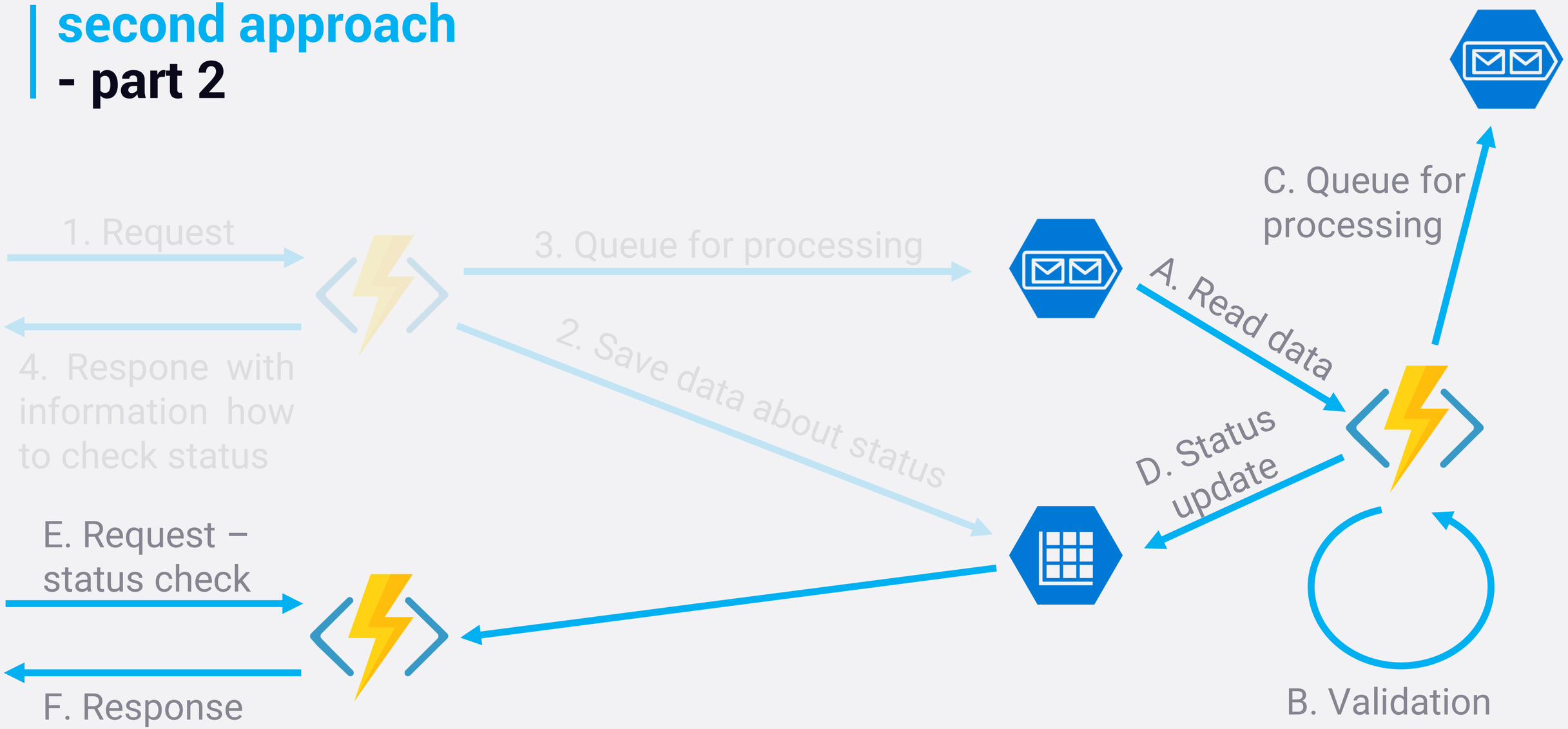


production solution II

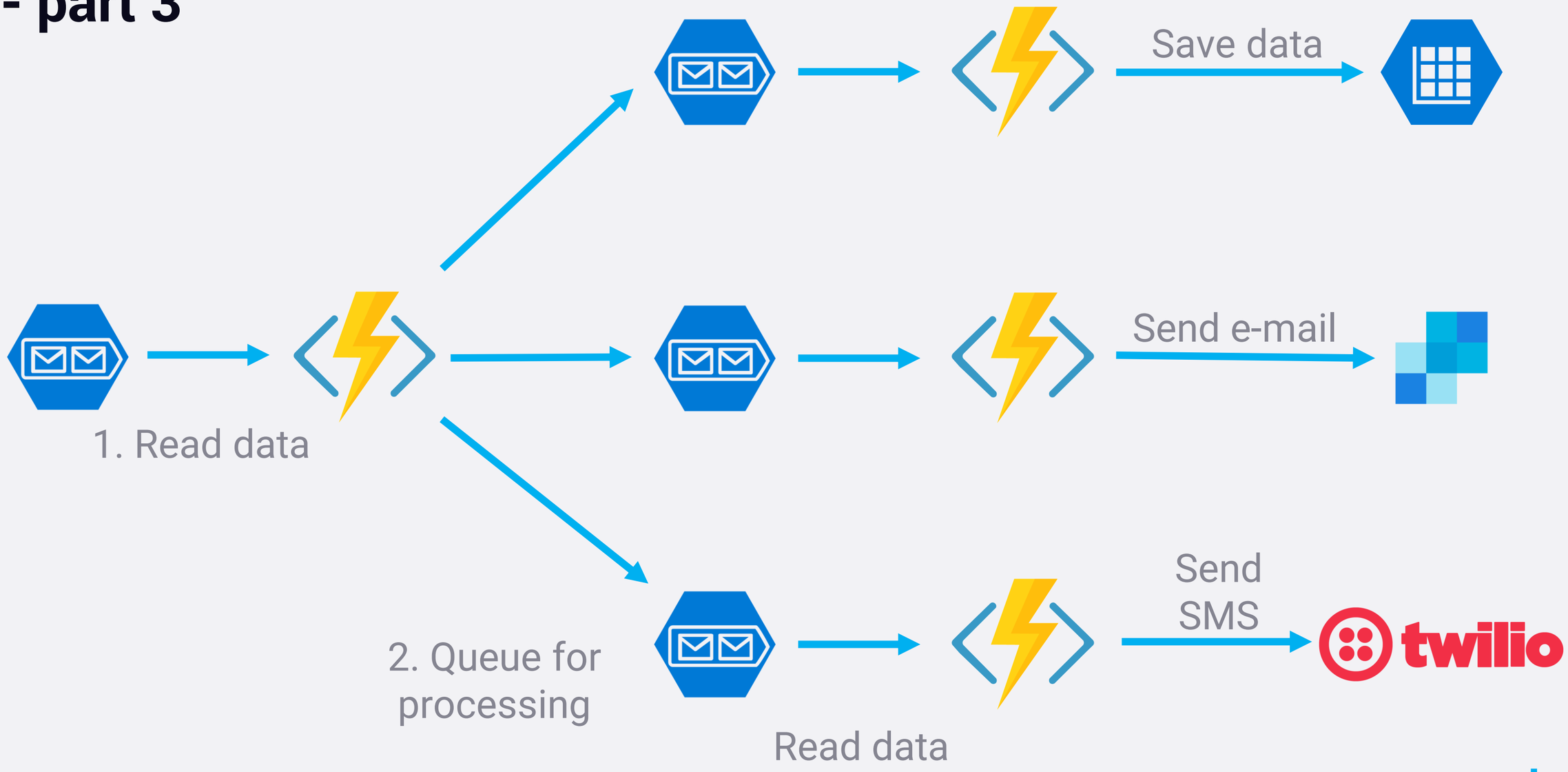
second approach
- part 1



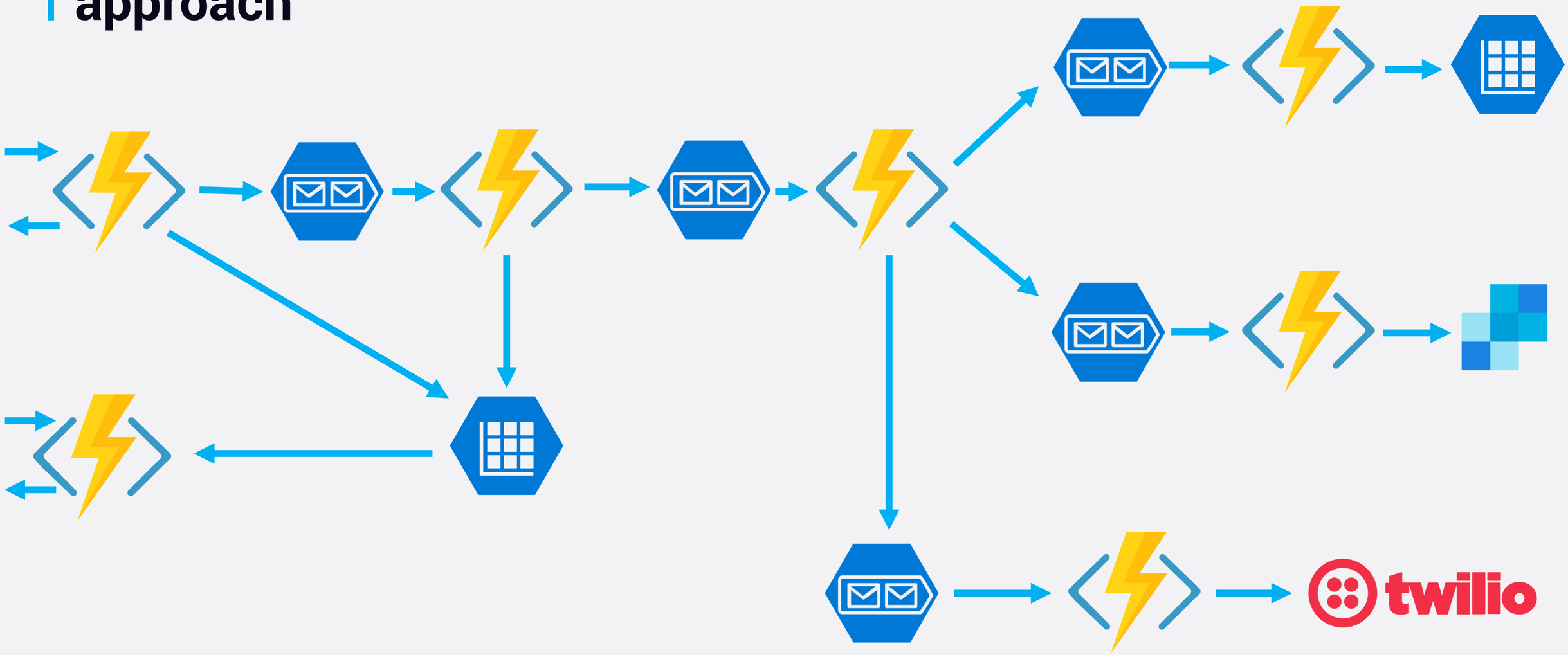
second approach
- part 2



second approach
- part 3

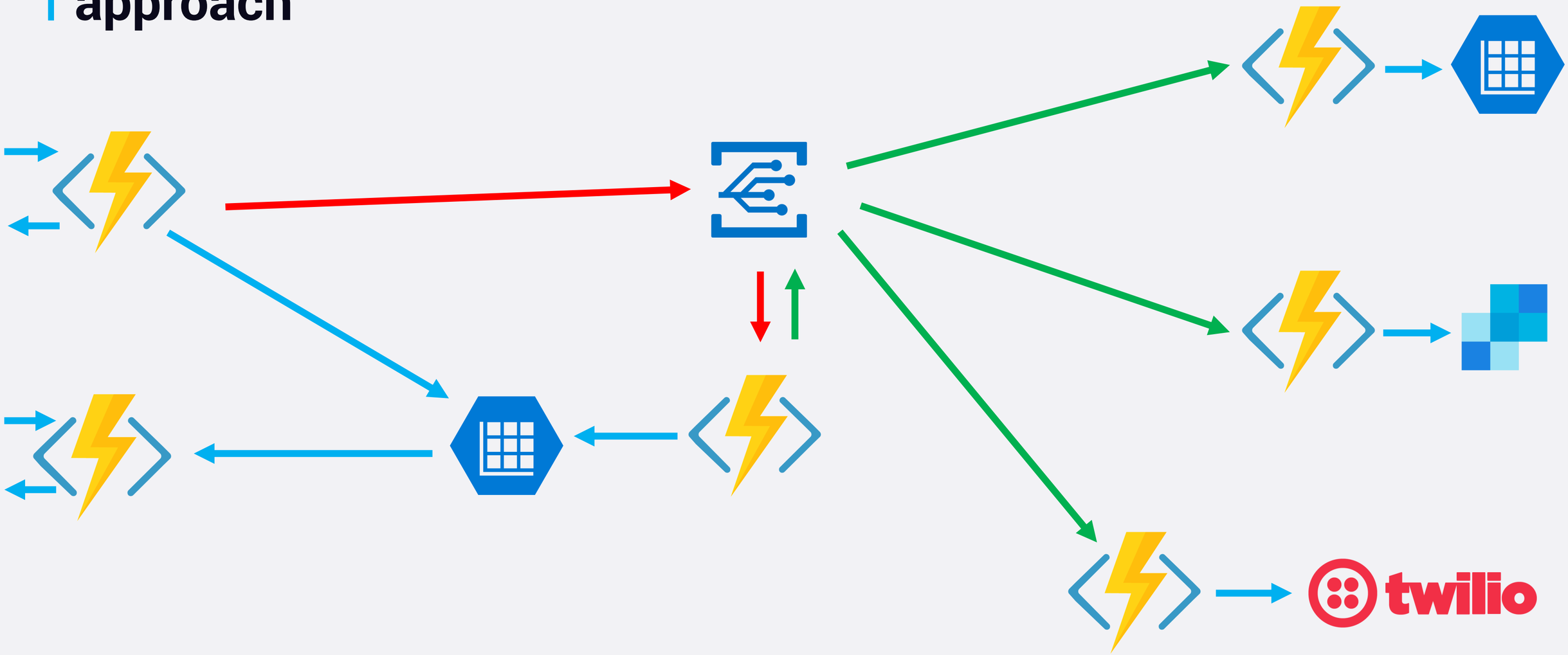


second approach



production
solution III

third approach



can you indicate
the best approach?

summary.

After today you should know what serverless term means.

You are aware of main concepts and how you should change your approach in solution design.

You know what the essential serverless components in Azure are.

do you have any questions?

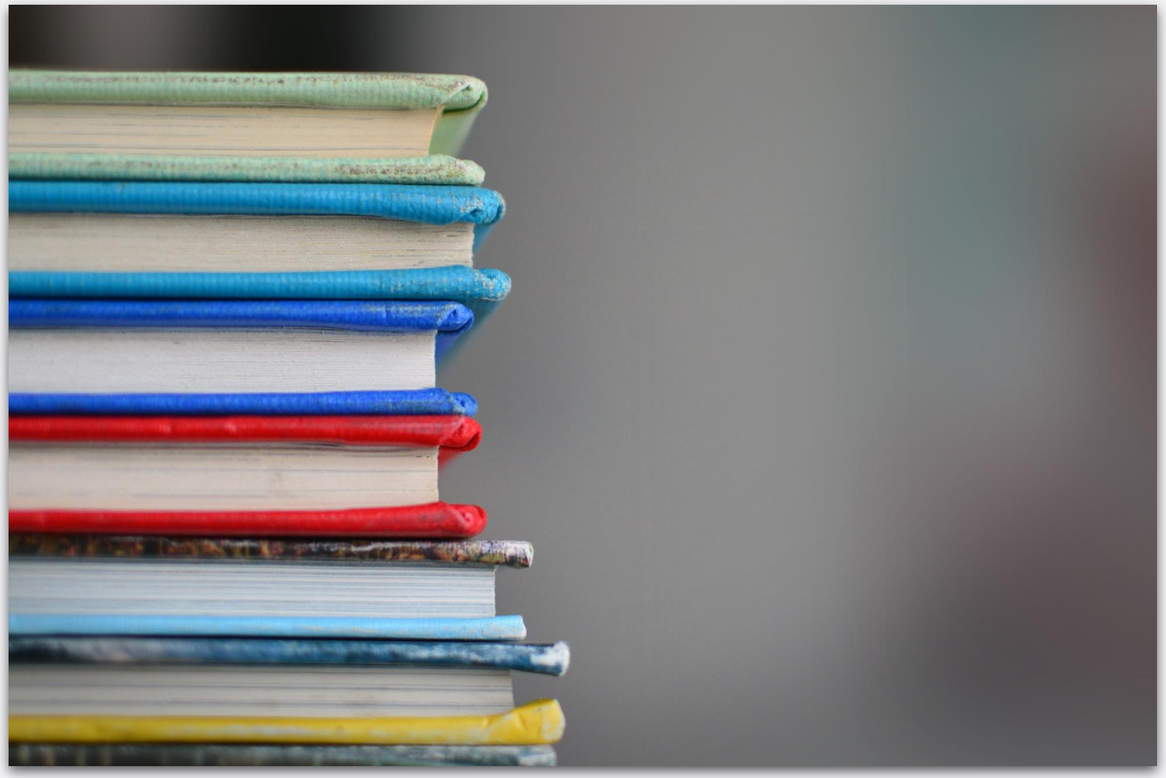
www.jankowskimichal.pl

mail@jankowskimichal.pl

@JankowskiMichal

github.com/MichalJankowskii

more information



- https://docs.microsoft.com/en-us/azure/azure-functions/
- https://github.com/Azure/azure-functions-core-tools
- https://github.com/Azure/Azure-Functions
- https://docs.microsoft.com/en-us/azure/event-grid/
- https://docs.microsoft.com/en-us/azure/logic-apps/
- https://github.com/Azure/azure-webjobs-sdk-extensions/

thank you

www.jankowskimichal.pl

mail@jankowskimichal.pl

@JankowskiMichal

github.com/MichalJankowskii