(영(0)(0)(어)[되 (C|L(0)|U|D)





Agenda

Date: 28.04.2022 Time: 17:30 Location: Talent Garden

17:45 - 18:00	Introduction
18:00 - 18:45	Topic #1 + Quiz : Serverless CI/CD with Cloud Build & Cloud Run, Alexander Hirz, Viktor Kiss
18:45 - 19:00	Break & Networking
19:00 - 19:45	Topic #2 + Quiz: The integrated company - automating business processes with GCP, Gökhan Sevilmiş, Muris Kavlak
19:45 - 20:30	Networking









Your reliable and competent partner in innovative cloud solutions and integrations

Born in the Cloud

- Founded in March 2020
- Google Cloud Partner since Mai 2020
- Focus
 - GCP Application- and Infrastructure Modernization
 - Google Workspace
- Advisor and Foundational work

- 18 Google Cloud Professional Certs
- 4 Google Cloud Authorized Trainer
- 50 Customers in DACH
- 12 Google Cloud Expertises
- 4 Public Customer Success Stories

"Our **mission** is to enable companies to focus on their core business by leveraging cloud native technologies."

"Our **vision** is a world where IT just works."



Your Speakers

Alexander Hirz

Cloud Architect, happtiq

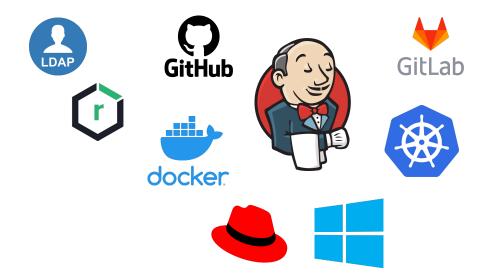




Viktor Kiss

Cloud Consultant, happtiq

Why serverless CI/CD?







- → Technology evolves faster and faster everyday
 - ightarrow Offers more possibilities
 - → More possibilities means more tools
 - → More tools means more complexity
 - → More complexity means more points of failure

Why serverless CI/CD?

Developers should focus on *development* not *infrastructure*

The plan:

1. Phase 1: Design

. Phase 2: Develop

3. Phase 3: ???

Phase 4:



Short Overview

Our "holy trinity" of CI/CD:

- Cloud Build
- Artifact Registry
- Cloud Run







Billing

- per build minute (partially consumed minutes are billed in seconds)
- "standard" machine type offers free-tier (120 build minutes / day)
- ingress & egress traffic covered separately

Storage options

- Artifact Registry
- Cloud Storage
- 3rd party repositories

Deployment options

- Google Kubernetes Engine
- Cloud Run
- Cloud Functions
- AppEngine
- Anthos
- Firebase

Features

- Build steps executed as Docker containers
- Several configuration options (Cloud Build as YAML / JSON, Dockerfile or Buildpacks)
- Build steps / images are open sourced (and backed by community)
- Accessible through REST API, Cloud Console or Google Cloud CLI
- Offline build available as well
- Version control systems easily integratable (others via Webhooks)
- private worker pools possible as well (offer more machine types)

Event triggers

- Push to a branch
- Push new tag
- Pull request
- Manual invocation
- Pub/Sub message
- Webhook event



Supported programming languages

Any: Write code your way using your favorite languages (Go, Python, Java, Ruby, Node.js, and any other running in a container)

Billing concept

- Only pay when your code is running for
- CPU (provisioned vCPU per 100 ms)
- Memory (provisioned GiB per 100 ms)
- Requests (per million requests)

Use Cases

Build a web app (web app, request-based services), developing APIs (web and mobile backends, internal APIs and services), data processing, automation (workflow and orchestration, event-driven reactive automation), connecting cloud services

Features

- Autoscaling based on concurrency (from 0 to 1000 or even more by support request)
- No billing if instances don't run
- Supports buildpacks & docker images
- No size limit to container images
- Multiple revisions (rollbacks & traffic splitting)
- Easy to setup CD with just a few clicks
- Deployment from source (Artifact Registry required)
- Access Control through IAM
- Automatically secured through HTTPS







Limitations

- 8 CPU & 32GB per instance
- Has to serve requests per HTTP(/2),
 WebSockets or gRPC
- Does not require local persistent file system
- Application has to be containerized



Supported types

Helm, Docker, Java, Python, Nodes.JS, APT, RPM

Billing

- Free tier (up to 0.5GB)
- Over 0.5GB -> 0.10\$ / GB
- Network ingress is free
- Network egress cost depend on repolection and traffic destination
- Traffic within same region is free

Features

- Supports a variety of artifacts
- Location / region customizable (latency)
- Can be used with all GCP compute services
- Access control per repo via GCP IAM
- Vulnerability scanner for container images (executed on push)

Use Case

Artifact Registry is a single place for your organization to manage container images and language packages (such as Maven and npm). It is fully integrated with Google Cloud's tooling and runtimes and comes with support for native artifact protocols.



Live Demo



Live Demo



Quiz









Cloud Functions vs Cloud Run





	Cloud Functions (v1)	Cloud Run
Code	Snippets (functions)	Container Images
Languages	Limited	Any
Duration	Up to 9min	Up to 60min
Requests	Only 1 at a time / instance	Multiple concurrent requests / instance

Automating Business Processes with GCP

Your Speakers

Muris Kavlak

Co Founder, happtiq





Gökhan Sevilmiş

Co Founder, happtiq

The Beginning

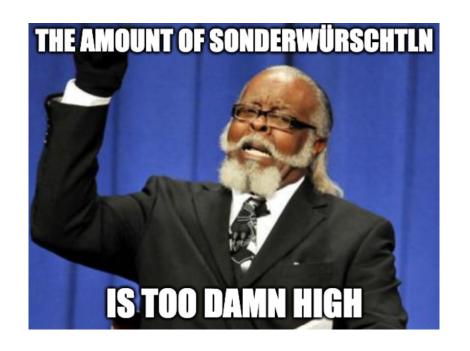
A consulting startup with

- Many tool requirements
 - o CRM, Invoicing, time tracking, ...
- The wish for a neatless solution
 - o Do it right from the start at least try
- Limited budget & no funding



Our Situation

- No "eierlegende Wollmilchsau" Tool
 - Expensive
 - o On-premise and not flexible
 - **But** pretty good standalone apps
- Automation, flexibility & extensibility
 - Billing & invoicing complexity
 - Specific customer requirements
 - o Reporting for sick leave, HO, ...
- Integration
 - o Background & masters of glue code
 - Functionality layer



Our Approach

"Don't scale your problem, scale your business"

The Result

- Technology & Services
 - Apps
 - Hubspot
 - Gmail, Drive, Chat
 - sevDesk
 - Clockify
 - askDante
 - o GCP & completely Serverless
 - Cloud Functions
 - Firestore
 - BigQuery
 - Cloud Build
 - Pub/Sub
 - Cloud Scheduler
 - Cloud Tasks
 - Secret Manager

- Automated Processes
 - Sales
 - Leads
 - Offers
 - New customer
 - Invoicing
 - Licensing
 - Servicing
 - Reporting
 - PTO
 - Homeoffice
- Full solution <u>Diagram Tool</u>

The Result

Without Automation

- Manual Effort
 - Sales
 - Scoping Calls
 - 3 min / lead
 - Offering
 - 5 min / deal
 - Invoicing
 - Licensing
 - 10 min / customer
 - Servicing
 - 5 min / customer

With Automation

- Click & Review
 - Sales
 - Scoping Calls
 - 0.5 min / lead
 - Offering
 - 1 min / deal
 - Invoicing
 - Licensing
 - 1 min / customer
 - Servicing
 - 1 min / customer

The Technology



Supported programming languages

Node.js, Python, Go, Java, .NET, Ruby, and PHP

Billing concept

- Only pay when your code is running for
- **Invocation** (first 2 million free per month)
- GB-seconds (provisioned memory per 100 ms)
- GHz-seconds (provisioned CPU GHz per 100 ms)

Use Cases

Automation (workflow and orchestration, event-driven, reactive automation: run function when photo is uploaded), application integration services (video and image analysis, sentiment analysis), serverless IoT backends, real-time file and stream processing, virtual assistants, and conversational experiences

Features

- Connect and extend Google Cloud services
- Single-purpose microservice
- Cloud event or HTTP request trigger
- End-to-end development and observability
- Avoid lock-in with open technology
- Retries

Regions

- Warsaw (europe-central2)
- Belgium (europe-west1)
- Frankfurt (europe-west3)
- London (europe-west2)
- **Zürich** (europe-west6)

The Technology

Functions framework

- Backbone library for Cloud Functions
- Open-sourced
- Allows for local development
- Transparent versions
- Deploy to other Clouds
- Transforms your container
 - Load the source code
 - Create an **HTTP server** (for Node.Js, example: Express 4)
 - Create a **route** for your request
 - HTTP, background functions, or CloudEvents





Live Demo



Quiz









It's Feedback time!









Learn more...









SAVE THE DATE! #3 Meetup - 30.5.2022







(영(0)(0)(어)[되 (C|L(0)|U|D)





Q&A









Herzlich Willkommen





