

Initial Report

Serverless Application

Rettaroli Andrea

July 2021

1 Introduzione

Nel 2016 il mercato delle architetture Serverless aveva un valore di 1,9 miliardi di dollari, nel 2020 il valore è arrivato a 7,6 miliardi e si prevede che nel 2021 arriverà a 21,1 miliardi. Ad oggi le più grandi aziende di tecnologie: Google, Amazon, Microsoft, all'interno delle rispettive piattaforme cloud: Google Cloud, AWS, Azure; hanno messo a disposizione servizi che permettono lo sviluppo secondo architetture Baas e Faas e hanno visto crescere notevolmente il loro fatturato grazie ad esse. Con l'avvento del cloud questa nuova forma di sviluppo e architetture ha preso fortemente piede perchè permette di scorporare le attività di routine per il provisioning, la manutenzione e la scalabilità dell'infrastruttura server che vengono gestite da un provider di servizi cloud. Gli sviluppatori devono semplicemente preoccuparsi di creare pacchetti di codice da eseguire all'interno di container remoti o di progettare le loro soluzioni includendo e sfruttando i servizi messi a disposizione dai cloud provider. Vedremo come questo nuovo paradigma di sviluppo distribuito influenza e cambia gli approcci della programmazione.

2 Obiettivi

L'obiettivo principale del progetto è quello di studiare e analizzare i concetti principali che si celano dietro il paradigma Serverless e di capirne e provarne ad utilizzare le architetture tipiche analizzandone le caratteristiche che hanno fatto sì che prendesse così tanto piede nello scenario tecnologico odierno. I punti su cui focalizzerò il mio

studio saranno:

- Architetture e concetti su cui si basa il paradigma Serverless;
- Vantaggi e svantaggi dell' approccio Serverless;
- Analisi dei principali servizi AWS quali Cognito, Lambda, DynamoDB, S3 e delle loro applicazioni all'interno di architetture Baas e Faas;
- Analisi dell'evoluzione di API REST in seguito all'integrazione di architetture Faas e Baas;
- Analisi e confronto con architetture e paradigma Paas.
- Creazione di applicazioni per scenari di test quali: autenticazione(registrazione, login, recupero password), CRUD, storage dati in c# per apprendere meglio i vari concetti.

3 Piano di lavoro

- **Piano teorico:** Studiare e comprendere come funziona il paradigma Serverless, quali sono le principali architetture; i vantaggi e gli svantaggi all'interno di casi di applicazioni reali. Analisi dei principali servizi messi a disposizione dai provider di servizi cloud.
- **Sviluppo:** Creazione di applicazioni Serverless che comprendano le principali operazioni che si svolgono nelle tipiche applicazioni web: autenticazione(registrazione, login, recupero password), CRUD, storage dati mediante l'uso delle architetture Faas e Baas, integrando logiche a micro-servizi con i servizi di AWS sopracitati.
- **Relazione finale:** Tutti i risultati dello studio verranno riportati in una relazione dettagliata.