**Lez1 22/05/2024**

**cloud service and container**

**cloud:** fare uso di un computer (server) e i servizi che offre ma non possederli a livello fisico

servizio costoso, molte variabili da considerare per la gestione dello

spazio/server/servizi gestiti

esempi cloud **AWS**(+diffuso, da 2005), **Azure**(dal 2009), e **GCP**

**Concetti del cloud**

**Virtualizzazione:** collegamento a server remoto, contenuto assieme ad altri server paralleli virtuali(isolati tra loro) all'interno di un server fisico unico(es macchine virtuali del corso)

**virtual private cloud:** stanza server privata = data center da riempire come si desidera. Non implica un server fisico, ma virtualizzato (ma comportamento simile)

**Scalabilità:**

**di volume:** funziona in base all’aumento/diminuzione traffico dell’utenza (cloud aiuta in questo, in modo che spazio utenza sia virtualmente illimitato)

**funzionale**: implementazione di servizi (gestito in programmazione non da cloud)

considera anche **Orizzontale**(lavoro alla volta) / **Verticale** (lavoro in parallelo)

<https://docs.aws.amazon.com/>

ricerche per materiale

**EC2**

<https://docs.aws.amazon.com/it_it/ec2/?icmpid=docs_homepage_featuredsvcs>

<https://docs.aws.amazon.com/it_it/AWSEC2/latest/UserGuide/concepts.html>

<https://docs.aws.amazon.com/it_it/AWSEC2/latest/WindowsGuide/concepts.html>

<https://docs.aws.amazon.com/it_it/AWSEC2/latest/APIReference/Welcome.html>

<https://docs.aws.amazon.com/it_it/cli/latest/reference/ec2/>

<https://docs.aws.amazon.com/it_it/ec2-instance-connect/latest/APIReference/Welcome.html>

**S3:** servizio aws per memorizzare file in bucket

<https://docs.aws.amazon.com/AmazonS3/latest/userguide/Welcome.html>

**Cloud9:** IDE online, fa girare codice con cloud non da nostro computer

<https://docs.aws.amazon.com/cloud9/latest/user-guide/welcome.html>

<https://docs.aws.amazon.com/cloud9/latest/APIReference/Welcome.html>

<https://docs.aws.amazon.com/cli/latest/reference/cloud9/>

**Lambda:**

<https://docs.aws.amazon.com/lambda/latest/dg/welcome.html>

**serverless** *application:* faccio eseguire ad amazon il mio app senza roa mia

[https://aws.amazon.com/](https://docs.aws.amazon.com/cli/latest/reference/cloud9/)getting-started/tutorials/run-serverless-code

?????

**Cloud computing:** strumento di calcolo, db, storage… su richiesta, via internet, ***Pays As You Go*** pricing (si paga in tempo utilizzo e qtà dati; costo a consumo)

Nel modello senza cloud porta a costi di personale, componenti(lungo hardware procurement cycle), pianificazione/organizzazione e sicurezza

Nel modello cloud possiamo pensare l’infrastruttura hardware a lvl software, è + flessibile, dinamica, a spese ridotte, task configurazioni uguali a tutti e perdita di tempo irrilevante(?)

**MODELLI di cloud**

*da + controllo delle risorse informatiche a minore controllo*

**IaaS (Infrastructure as a Service)**

come avere server/hardware, tutto a mio carico per gestione lato software

**PaaS (Platform as a Service)**

no accesso al server, responsabile a lvl configurazione software ma ridotto

**SaaS (Software as a Service)**

prodotto che configuro per utilizzo

Oggi è tutto SaaS

**Tipi di cloud**

**Full Cloud:** tutta infrastruttura e servizi in cloud(accesso online)

**Hybrid:** mix tra solo cloud (servizi) e premises(dati tenuti per sé in db privati per es).usati da banche tendenzialmente

**Private Cloud(on-premises):** cloud a data center privato, (non usano server pubblici, tutto il “cloud” è di proprietà)

**Cloud computing**

**Economia di scala** (vantaggi)

tutto automatizzato, azioni prodotte in massa , costi ridotti.

**Riduzione spese = Investimento nei servizi per clienti**

**Efficienza / Capacità / Scalabilità** meglio gestite

**Incremento velocità, prestazione**

**Attività trasferibile/utilizzabile a lvl globale in poco tempo**

**Container:** (?!)blocco di codice che lavora, pagamento da fare per la qtà dei componenti usati (scelti da utente / classi di prezzo)

cit meccanismo **request / response** , anch’esso è gestito dal cloud

AWS è una piattaforma che fornisce servizi in caso di necessità, flessibile e l’user paga in base al tempo di utilizzo.

visualizzazione delle **macrocategorie** dei servizi (backend e frontend)

esempio creazione app:

networking(VPC) > Compute(EC2) > Database(Dynamodb) > storage(S3)

**Compute:** server per caricare codice e sia visualizzabile (es caricare web application o altro) (contiene 10 servizi)

**Storage**

**Database**

**Networking**

**Management and Governance**

**Security**

**Cost Management**

**metodi connessione con aws:**  management console (interfaccia grafica ), CLI(command line interface), Software Development Kits (SDK: pacchetti per lavorare da codice)

**AWS Cloud Adoption Framework (AWS CAF):** in base a prospettive di

[***Business***, ***People***, ***Governance*** (:prospettiva business)]

[***Platform***, ***Security***, ***Operations*** (:prospettiva tecnica)]

per creare progetti….