# 11- DOCKER

Docker è una paittaforma sw che cerca di semplificare il rpcoesso di costruzione, gestione, deployment delle applicazioni → cerca di virtualizzare il sistema operativo in maniera più efficiente.

Scritto in linguaggio GO.

Nella maggior parte dei casi, evita problemi di configurazione perché basta importare l’immagine del docker.

Container: entità logiche gestite dal docker host

* Ogni container ha una copia virtuale di
  + Process table
  + Interfacce di rete
  + File system
* Si effettua uno sharing delle risorse con l’applicazione che usa le risorse.

Immagine che contiene testo, schermata, Carattere, design

Descrizione generata automaticamente

Con VM: problemi di RAM ed etc.

Con Docker, si installa un docker engine sul quale installo direttamente tutte le varie immagini che saranno in grado di condividere le risorse della macchina.

**Vantaggi di Docker:**

Immagine che contiene testo, Carattere, schermata, algebra

Descrizione generata automaticamente

Basta effettuare una volta la configurazione e   
poi posso mandarla a tutti

Svantaggi:

* Difficoltà di configurazione
* Learning curve per utenti
* Supporto limitato per alcuni OS
* Non può avere sullo stesso host applicazioni basate su sistemi operativi differenti

DOCKER ENGINE: Crea e gestisce le varie immagini, container, reti, volumi

Formato da:

* **Server**: ci permette di interagire
* **Client**: da linea di comando
  + **Permette di dialogare con l’engine attraverso la linea di comando**
* **REST** **API**

Distinzione:

* **Immagine docker**: scheletro che creo una volta, contenente tutta la configurazione (es: funzione)
  + statiche
* **Cotnainer docker**: istanza dello scheletro che eseguono l’applicazione contenuta nell’immagine
  + Dinamici

DOCKER HUB:

* Repo online per immagini docker
* Permette di storare e distribuire immagini customizzate
  + Da opzione per farla pubblica o privata
  + ---
  + ---

Immagine che contiene testo, schermata, diagramma, linea

Descrizione generata automaticamenteUn’alternativa è GitHubContainerRegistry ma non ci interessa

Nell pratica, quando creo un file di configurazione docker, metto istruzioni su ogni riga di questo gfile di configurazione. Ogni volta creo un layer read-only. In cima, c’è u

Immagine che contiene testo, Carattere, schermata, bianco

Descrizione generata automaticamente

Nota: se io ho caricato un file e poi lo cancello → rimane salvato sia l’inserimento che la cancellazione in quanto ogni volta che si fa un’istruzione → si aggiunge un layer.

I vari layer sono read-only, non possono essere modificati → in base alle configurazioni del fyle system posso creare un layer ed invalidare i layer sottostanti.

Immagine che contiene testo, schermata, Carattere, numero

Descrizione generata automaticamente

Alla fine, c’è un layer finale read-write che posso editare, copiare.

LE varie modifiche sono salvate su questo finale

Immagine che contiene testo, schermata, Carattere, design

Descrizione generata automaticamentePosso avere diverse immagini, anche che estendano altre. Il livello del container, viene eliminato quando distrigguo il docker container.

Attenzione: il docker container è effimero → quando lo faccio ripartire, perdo quello che ho fatto prima, devo salbavre i miei progressi in qualche altro modo.

Posso condividere tutti i vari layer read-only nei vari container ache usano la stessa immagine; l’unico ad essere distinto sarà l’ultimo layer che è read-write.

Immagine che contiene testo, schermata, diagramma, Blu elettrico

Descrizione generata automaticamente

Immagine che contiene testo, Carattere, schermata, bianco

Descrizione generata automaticamentePer far resistere le modifiche → devo storare nel filesystem, isolandoli dal container.

Docker file: file di testo che contiene tutte le istruzioni per costruire la nostra immagine:

* Specificare immagine da cui si parte
* Specificare tutti i vari comandi che servono per buildare la nostra applicazione

Immagine che contiene testo, schermata, Carattere

Descrizione generata automaticamente

Dopo di ciò posso creareun container basato su quel file:

Immagine che contiene testo, Carattere, bianco

Descrizione generata automaticamente

→ mi espoinde sulla porta 3000 del mio container, quello che è stato eseguito (porto da 3000 del container, a 3000 della mia macchina) → forward della porta 3000 del mio docker sulla porta 3000 del mio host.

**Docker Compose:**

file di configurazione contenente i servizi:

* Container che esegue un ‘app
* Immagine che contiene testo, Carattere, schermata, bianco

  Descrizione generata automaticamenteOgni servizio rappresenta diversi componenti di una stessa applicazione.
* Usato nel caso in cui un solo container non è abbastanza e ho bisogno di più container.

Dunque la docker-compose.yml ha: l’iimagine da usare, le porte da esporre e le variabili di sistema.

Immagine che contiene testo, schermata, software, schermo

Descrizione generata automaticamente

client:

server:

* Volume: dove deve essere hostato

Network:

* Driver: bridge → faccio in modo che server e client possano dialogare tra di loro attraverso quella porta

Immagine che contiene testo, Carattere, schermata, bianco

Descrizione generata automaticamente

**BUILD** → genera configurazione

**LAUNCH**→ fa partire i container

Immagine che contiene testo, Carattere, schermata, ricevuta

Descrizione generata automaticamenteConviene lanciare anche i test usando docker in quanto loro faranno in questo modo.

**Continuous Integration**

CI integra continuamente i cambiamenti di codice in una repo condivisa, seguita autonomamente da test e build.

Es: ogni volta che viene fatto qualcosa sul gitlab parte la pipeline di contiuous integration che controlla i test.

CD: Contiuos Deployment

Quando faccio una push, oltre alla pipeline di CI, si effettua un deploy in un ambiente di test

Immagine che contiene testo, schermata, Carattere, algebra

Descrizione generata automaticamente

* Aiuta a mandare in produzione codice sempre testato

Immagine che contiene testo, schermata, Carattere, linea

Descrizione generata automaticamente

Nota: bisogna fare in modo che solo i branch più stabili siano configurati in questo processo altrimenti ad ogni commit ricaricherà tutto.  
Noi ci concentriamo su Continuous Integration.

Immagine che contiene testo, schermata, Carattere

Descrizione generata automaticamente

Deve essere indipendente dal pc

→ far funzionare solo sul branch selezionato

Immagine che contiene testo, schermata, diagramma, Parallelo

Descrizione generata automaticamente

Immagine che contiene testo, schermata, Carattere, numero

Descrizione generata automaticamenteDi default tuta la pipeine di CI viene lanciata da un gitpush.

Runner sono dei processi che fanno partire tutto, cercano test e verificano tutto.

EZ ELETRONICS:

Applicazione react

Immagine che contiene testo, schermata, Carattere

Descrizione generata automaticamenteClient ha diverse pagine, molto minamali che popolano i dati che il server fornisce.

Npm install → per scaricare tutte le dipendenze.

Immagine che contiene testo, Carattere, bianco, schermata

Descrizione generata automaticamente

Nelle dipendenze, vanno scritte in modo tale che facendo npm installa, queste venfono installate correttamente.

Nodemon server.mjs/ npm run

Immagine che contiene diagramma, schermata, linea, giallo

Descrizione generata automaticamente

COME’è FATTO IL CODICE DEL SERVER:

* Index → pagina principale
* Jest → per i test
* Src → contiene i vari sorgenti
  + Contiene altre cose quali componenti, db, errori, routers…
* Test\_official → test dei prof
* Altri test → nostri

Nota: la persistenza non è stat già implementata → consigliato sqlite3

Successivamenre alla parte di sviluppo si effettuano i test:

* Unitari .→ singola funzione
* Integration → metti insieme più funzioni
  + Se aggancio anche il db, devo fare in modo che quando un test viene eseguito, deve lasciarmi nella sitauzione insiziale → un test non deve modificare lo stato del sistema
  + Ogi test deve essera tomico → deve partire sda uno stato ben definitio e deve portare il sitema allo stesso modo in cui l’avevo trovato
    - Si può usare un db alternativo per i test → uso una variabile d’ambiente
* End-to-end → uso front-end con browser e vedo se tutto funziona
  + Esistono degli script per fingersi utente e interagisocno con la pagina web in modo automatica

Ma se cambia la pagina web, vanno riscritti

* + - Selenium

Immagine che contiene testo, Carattere, schermata

Descrizione generata automaticamente