

Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA

CORSO DI LAUREA IN INFORMATICA



Containerizzazione di Malware Dashboard

Relazione finale di stage

Relatore

Prof. Tullio Vardanega

Laureando

Andrei Petrov

ANNO ACCADEMICO 2016-2017

Sommario

//TODO

Indice

1	L'Azienda	1
1.1	IKS	1
1.2	Profilo dell'azienda	2
1.2.1	Servizi e prodotti offerti	2
1.2.2	Struttura organizzativa	6
1.2.3	Processi aziendali	6
1.3	Rapporto con l'innovazione	8
2	Stage	11
2.1	Il valore aggiunto di uno stagista	12
2.2	Alcuni temi di stage	12
2.2.1	AIOps e Machine Learning	12
2.2.2	DevOps Automazione	12
2.2.3	Sviluppo moduli evolutivi in ambito antifrode	12
2.3	Il progetto proposto	13
2.3.1	Motivazioni	13
2.3.2	Obiettivi aziendali	15
2.3.3	Obiettivi personali	15
2.4	Piano di lavoro	16
2.5	Vincoli	17
2.5.1	Vincoli temporali	17
2.5.2	Vincoli tecnologici	17
3	Lo svolgimento dello stage	19
3.1	Metodo di lavoro	19
3.2	Attività di formazione	19
3.3	Analisi dei requisiti	19
3.3.1	Classificazione dei requisiti	19
3.3.2	Requisiti	19
3.3.2.1	Funzionali	19
3.3.2.2	Non funzionali	19
3.4	Progettazione e realizzazione	19
3.4.1	Scelte progettuali	19
3.4.2	Visione architetturale a microservizi	19
3.4.3	Codifica	19
3.4.4	Test	19
3.4.4.1	Test di carico	19
3.4.4.2	Test di durata	19

4	Valutazioni retrospettive	21
4.1	Obiettivi raggiunti	21
4.2	Problematiche riscontrate	21
4.3	Bilancio formativo	21
4.3.1	Il prima	21
4.3.2	Il dopo	21
4.4	Valutazione critica del Corso di Laurea	21
	Glossario	23
	Acronimi	25
	Riferimenti	27

Elenco delle figure

1.1	Visione a processo della gestione del rischio. Immagine tratta da: http://bit.ly/2rh3V0A	2
1.2	Flusso di lavoro durante lo svolgimento di un audit. Immagine tratta da: http://bit.ly/2rdFhfv	2
1.3	Visione grafica del concetto di difesa perimetrale. Immagine tratta da: http://bit.ly/2s834O2	3
1.4	Visione del ciclo di vita del processo business continuity. Immagine tratta da: http://bit.ly/2qvCmgP	3
1.5	Vista a confronto tra un ambiente server bare metal e virtualizzato. Immagine tratta da: http://bit.ly/2qvtLLk	4
1.6	Visione della gestione di servizio in prospettiva del Framework ITIL . Immagine tratta da: http://bit.ly/2qvNryk	4
1.7	Visione architetturale a monolite e microservizi a confronto. Immagine tratta da: http://bit.ly/2rh1niY	5
1.8	Organigramma aziendale	7
1.9	Rappresentazione grafica del coinvolgimento del Cliente e i corrispettivi livelli degli interventi del gruppo commerciale, tecnico, direzionale e di supporto nella gestione di un'offerta di progetto.	8
1.10	Il legame attivo tra innovazione e il processo del cambiamento e gestione della conoscenza. Immagine tratta da: http://bit.ly/2qty3XD	9
2.1	Visione d'insieme dell'approccio DevOps. Immagine tratta da: http://bit.ly/2seXw3o	13
2.2	Vista a confronto del metodo classico di virtualizzazione basato su macchine virtuali con il nuovo metodo di virtualizzazione leggera basato su container. Immagine tratta da: http://dockr.ly/23gH9Ad	14

Elenco delle tabelle

Capitolo 1

L'Azienda

Il presente capitolo è dedicato alla presentazione dell'Azienda che mi ha ospitato nella propria sede per uno stage obbligatorio per la Laurea triennale.

Ho strutturato la presentazione in tre parti: nella prima parte introduco l'azienda, nella seconda parte descrivo il suo profilo aziendale e nello specifico il settore di mercato in cui essa opera, chi sono i suoi clienti, come è strutturata e qual è il suo modello di qualità. Infine, descrivo il suo rapporto con l'innovazione.

1.1 IKS

IKS (*Information Knowledge and Supply*) è un'azienda padovana fondata dall'attuale Amministratore Delegato Paolo Pittarello nel 1999.

Nell'insieme, IKS unisce figure di alto profilo per proporre soluzioni innovative alle richieste di mercato dell'[Information and Communication Technology \(ICT\)](#) sia italiano che estero. Le soluzioni offerte interessano in particolare gli ambiti della sicurezza, dell'infrastruttura e della governance [Information Technology \(IT\)](#).

L'azienda è in continua ricerca tecnologica. Investendo sulla formazione del proprio personale, IKS si impegna di portare solo valore aggiunto al business dei propri clienti. Inoltre, l'Azienda crede fortemente nell'innovazione come strumento verso un ambiente digitale comune, [Agile \(metodologia\)](#) e completamente disponibile.

Il quartier generale aziendale è a Padova. Inoltre, IKS possiede uffici anche nelle seguenti città: Roma, Milano e Trento.

A partire dallo scorso anno, 2016, IKS SRL, Kirey SRL, Insirio SPA e System Evolution SRL hanno fondato il Gruppo Kirey. L'obiettivo comune delle quattro aziende è unire le competenze complementari e garantire un portfolio completo di soluzioni ai clienti attuali e futuri.

La creazione del Gruppo Kirey è stata guidata dalla Synergo SGR una società di [Private equity](#). E il presidente del nuovo Gruppo commerciale è Vittorio Lusvarghi.

A seguito della creazione del Gruppo, IKS SRL e le restanti tre realtà aziendali hanno conservato la propria struttura di governance e management, garantendo la continuità gestionale.

1.2 Profilo dell'azienda

1.2.1 Servizi e prodotti offerti

Nel corso degli anni IKS si è fatta notare per gli enormi contributi innovativi nell'ambito della sicurezza informatica. Tuttavia, essa non è limitata a questo ambito. Infatti, gli altri ambiti di applicazione sono: infrastruttura e governance IT.

Di seguito vengono presentati quali sono i servizi offerti da IKS per ciascun ambito:

* IT Security

– Risk analysis e vulnerability assessment

E' importante garantire la sicurezza dell'infrastruttura informatica nel suo complesso. A questo scopo IKS offre un servizio orientato alla ricerca di eventuali vulnerabilità e analisi dei rischi a esse collegate;



Figura 1.1: Visione a processo della gestione del rischio. Immagine tratta da: <http://bit.ly/2rh3V0A>.

– Audit management

Le aziende di continuo sono sottoposte a controlli di vario genere per accertare che le operazioni aziendali sono a norma con certificazioni, leggi, bilanci ed ecc. A questo scopo IKS offre un servizio di supporto per le aziende con il fine di agevolare le attività di auditing e eventualmente per migliorare i processi interni aziendali;



Figura 1.2: Flusso di lavoro durante lo svolgimento di un audit. Immagine tratta da: <http://bit.ly/2rdFhfv>.

– **Difesa perimetrale**

Sempre in ambito della sicurezza è importante prendere le giuste misure per garantire a priori uno specifico livello di sicurezza e limitare a zero le intrusioni dall'esterno di un'infrastruttura IT aziendale. A questo scopo IKS offre un'insieme di soluzioni orientate al monitoraggio degli accessi a sistemi aziendali, dei permessi sulle operazioni che un utente può fare e/o vedere, e molto altro;



Figura 1.3: Visione grafica del concetto di difesa perimetrale. Immagine tratta da: <http://bit.ly/2s834O2>.

* **IT Infrastructure**

– **Business continuity**

In ambito bancario, le infrastrutture informatiche sono molto complesse. E la loro manutenzione non è semplice. Ancora più difficile è garantire che questi sistemi siano operativi al 100%. Una simile percentuale nella pratica è impossibile. IKS con il proprio gruppo di esperti sono alla continua ricerca di soluzioni per incrementare la percentuale di continuità operativa di questi sistemi. Infatti, le soluzioni offerte dall'azienda sono orientate nel concreto all'infrastruttura del cliente richiedente supporto;



Figura 1.4: Visione del ciclo di vita del processo business continuity. Immagine tratta da: <http://bit.ly/2qvCmgP>.

– Virtualization technology

Ogni prodotto software per portare valore aggiunto a un'azienda deve essere eseguito. Mappare in esecuzione un prodotto software per server fisico richiede la disponibilità di un cospicuo numero di questi server. A questo scopo la tecnologia di virtualizzazione permette la creazione di server logici che eseguono programmi e a loro volta vanno in esecuzione su server fisici. I benefici di una simile infrastruttura è l'ottimizzazione delle risorse di calcolo, agilità di gestione e sicurezza. Alcune delle soluzioni di virtualizzazione offerte da IKS sono: VMWare, RHEV. Un'evoluzione della tecnologia di virtualizzazione è il cloud. In questo ambito IKS offre soluzioni di migrazione e supporto verso il Cloud dell'infrastruttura IT classica di un'azienda;

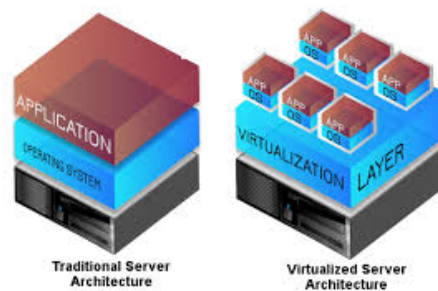


Figura 1.5: Vista a confronto tra un ambiente server bare metal e virtualizzato. Immagine tratta da: <http://bit.ly/2qvtLLk>.

* IT Governance

– Service management

Un servizio informatico richiede costante attenzione. Questo necessita di enormi investimenti economici per la manutenzione. IKS offre diversi piani di gestione per soddisfare anche i più esigenti clienti;

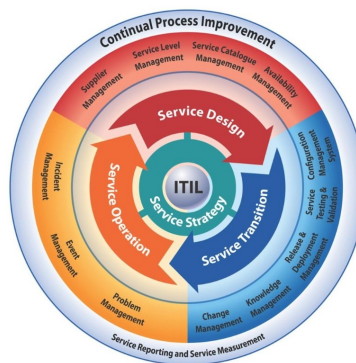


Figura 1.6: Visione della gestione di servizio in prospettiva del Framework ITIL. Immagine tratta da: <http://bit.ly/2qvNryk>.

– **Application and performance monitoring**

Un prodotto software ha il proprio specifico ciclo di vita. Concluso il ciclo di sviluppo, il prodotto è rilasciato in produzione. La seconda parte del ciclo di vita di un prodotto software è la manutenzione. Essere in grado di monitorare un applicativo è importante per avere una costante visione dello stato del prodotto e prevenire eventuali esigenze di manutenzione generica oppure di basso profilo a livello di codice sorgente. In questo dominio, grazie a partnership strategiche, IKS offre soluzioni mirate a garantire la miglior possibile esperienza di monitoraggio applicativo;

– **System and networking management**

Gestire sistemi e reti complessi è un compito complesso. Utilizzare strumenti adeguati permette di semplificare il lavoro e garantire un stato consistente del sistema nel tempo. Le soluzioni offerte di IKS sono orientate alla flessibilità e facilità d'uso dei prodotti offerti in questo contesto;

* **Innovation & Project**

– **Architetture applicative distribuite**

I sistemi informatici diventano sempre più di natura distribuita. IKS offre in questo ambito soluzioni architetturali orientate a microservizi, utilizzando le ultime tecnologie orientate alla containerizzazione e orchestrazione di container;

– **Sviluppo di applicazioni cloud native**

E' sempre più comune sentire parlare di cloud. Le classiche architetture applicative non riescono a beneficiare della flessibilità del cloud perché non sono scalabili e sono difficilmente modularizzabili vista la loro architettura a monolite. Per questo motivo le applicazioni devono essere sviluppate fin dal principio con un'architettura orientata al Cloud. Una buona guida di sviluppo di applicazioni web orientate al Cloud è la *Twelve Factor-App*. In questa direzione IKS si impegna di proporre soluzioni architetturali orientate all'affidabilità, resilienza, scalabilità orizzontale ed ecc.

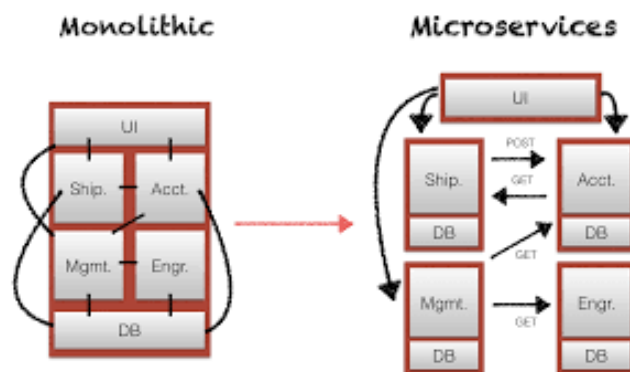


Figura 1.7: Visione architetturale a monolite e microservizi a confronto. Immagine tratta da: <http://bit.ly/2rh1niY>.

La clientela tipica di IKS sono aziende che operano nell'ambito della pubblica amministrazione, bancario, assicurativo e servizi.

Una lista dettagliata delle referenze può essere consultata sul sito di IKS all'URL <https://www.iks.it/referenze.html>.

1.2.2 Struttura organizzativa

Ad oggi IKS conta più di 100 dipendenti. La sua organizzazione interna è riassunta nel diagramma in **Figura 1.8**.

In seguito descrivo le unità operative che costituiscono il nucleo decisionale dell'azienda. Queste unità sono:

- * **Direzione**
Definisce gli orientamenti e le politiche aziendali. Definisce gli obiettivi per la qualità, riesamina periodicamente il sistema di qualità e gestisce il piano di formazione dei dipendenti in funzione alle esigenze e motivazioni personali;
- * **Direzione Commerciale**
Definisce le politiche commerciali, gli obiettivi e le risorse necessarie. Promuove i servizi e prodotti dell'azienda. Gestisce i clienti, i fornitori e le offerte contrattuali;
- * **Direzione tecnica o Operation**
Supporta la Direzione Commerciale nella valutazione commerciale di prodotti e/o offerte dal punti di vista tecnico. Gestisce a livello tecnico i progetti e servizi. Pianifica le risorse necessarie per i prodotti/servizi. Verifica lo stato del prodotto/servizio offerto;
- * **Amministrazione & Finanza**
Gestisce la documentazione di progetto su coordinamento della direzione commerciale e tecnica. Gestisce l'archiviazione della documentazione;
- * **Acquisti**
Su coordinamento della Direzione, gestisce i fornitori di prodotti e servizi. Gestisce il processo di acquisizione di nuovi prodotti o servizi. Il processo di acquisizione è guidato dalle necessità interne aziendali oppure da quelle dei clienti;
- * **Assicurazione Qualità**
E' in stretto contatto solo con la Direzione. Gestisce il piano di qualità, coordina le attività di ispezione, misura e stima il livello della qualità aziendale;
- * **Business Unit (BU)**
Gestisce i progetti o servizi concordati con il Cliente. Rendiconta direttamente alla Direzione Tecnica e gestisce l'emissione delle fatture verso il Cliente.

1.2.3 Processi aziendali

IKS dal 2003 è certificata UNI EN ISO 9001. Questo certifica che l'azienda cura molto la qualità del lavoro interno. Infatti, migliorare di continuo il proprio *modus operandis* permette all'azienda di rimanere competitiva sul mercato e consolidare la propria posizione di leader sul mercato del ICT italiano. Riporto di seguito alcuni obiettivi di qualità dell'azienda:

- * Mantenere e aumentare il livello di soddisfazione del Cliente;

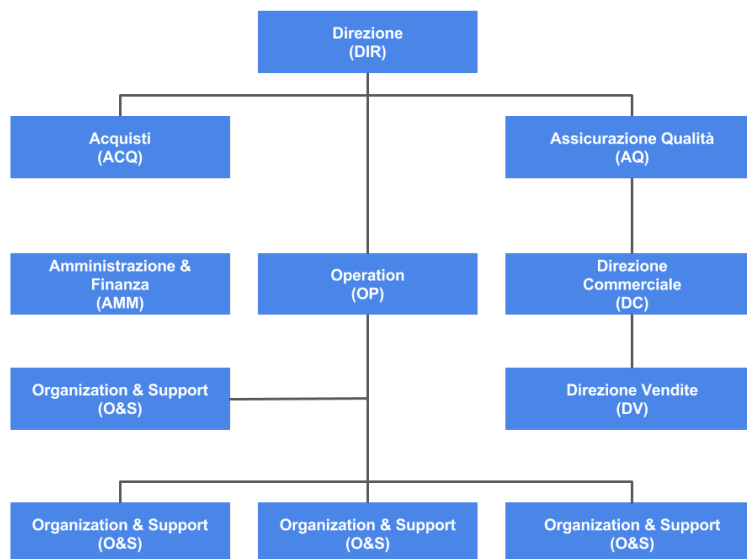


Figura 1.8: Organigramma aziendale

- * Operare in modo efficiente ed efficace per soddisfare i requisiti contrattuali, norme e regolamenti;
- * Monitorare i propri processi per garantire azioni correttive tempestivamente e permettere un comportamento pro attivo, per anticipare i bisogni e predire le risorse aziendali necessarie prima del loro effettivo bisogno;
- * Assicurare una adeguata formazione al Personale.

Il ruolo del Cliente è di prima importanza. Infatti, l'azienda cerca di coinvolgerlo il più possibile per comprendere meglio i suoi bisogni. Dopo aver formalizzato il bisogno del Cliente segue un'attività di analisi dei requisiti. L'obiettivo dell'attività è la dettagliata comprensione del contesto applicativo, quali sono le parti interagenti e quali possono essere i rischi durante l'attività di progetto per implementare i bisogni del Cliente.

A progetto concluso, il Cliente valuta criticamente la soluzione presentata. La valutazione può coinvolgere anche un reclamo, il quale è rivolto alla Direzione dell'azienda.

IKS organizza il proprio lavoro per processi: primari, direttivi e di supporto. Ciascuna categoria di processo definisce delle responsabilità e compiti. Per esempio i processi organizzativi interessano le attività per: definire la politica e strategia aziendale, pianificare e allocare le risorse, riesaminare la gestione del sistema di qualità. Invece, i processi primari ricoprono attività che garantiscono un diretto ricavo economico per l'azienda e danno un valore aggiunto al prodotto o servizio fornito. Esempi di attività sotto questa categoria sono: proporre offerte commerciali ai clienti, progettare e sviluppare prodotti software, erogare servizi IT. L'ultima categoria di processi sono quelli di supporto. Tra le consuete attività giornaliere rientrano le seguenti, per esempio: gestire le risorse umane, l'infrastruttura e gli ambienti di lavoro, monitorare e analizzare la qualità aziendale.

Una rappresentazione grafica di come sono legate le parte durante un progetto segue in **Figura 1.9**.

Periodicamente il responsabile della qualità su mandato della Direzione attua attività d'ispezione per controllare la qualità. A posteriori segue un'attività di analisi e misura i livelli di qualità per ogni reparto interno, servizio e prodotto di IKS.

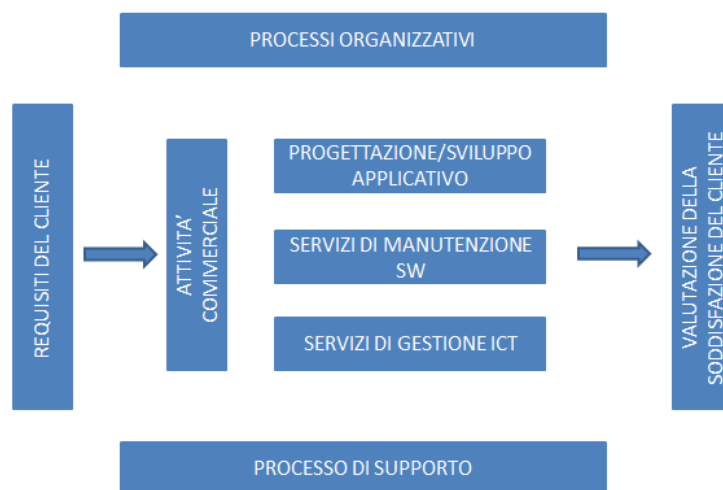


Figura 1.9: Rappresentazione grafica del coinvolgimento del Cliente e i corrispettivi livelli degli interventi del gruppo commerciale, tecnico, direzionale e di supporto nella gestione di un'offerta di progetto.

1.3 Rapporto con l'innovazione

L'innovazione è il processo di gestione dell'intero ciclo di vita di un'idea. L'obiettivo è portare un miglioramento di processo aziendale, di prodotto e/o di servizio. Il seguente miglioramento si traduce in un valore aggiunto per l'azienda in termini di rientro economico e per il Cliente nel soddisfare un bisogno in modo più efficace ed efficiente.

L'approccio innovativo induce l'utilizzo dell'informazione, della creatività e dello spirito d'iniziativa per raccogliere maggior valore aggiunto dalle risorse a disposizione. L'azienda utilizza l'innovazione per soddisfare in modo pro attivo le richieste del Cliente. Questo principio è pienamente in linea con la strategia di qualità aziendale: *client first*.

La modalità di innovazione di IKS è un approccio incrementale. Inizialmente l'azienda cerca di soddisfare i bisogni principali e raggiungere il prima possibile gli obiettivi fissati. In seguito, l'azienda si impegna di migliorare la propria offerta mediante miglioramenti continui a ogni livello di dettaglio.

Per supportare l'innovazione IKS ha creato una cultura aziendale che permette ai propri dipendenti di scambiarsi idee, sperimentare, imparare in gruppo e mettere in atto la propria creatività. Non manca la comunicazione con i propri responsabili. Questi sono i primi a motivare di continuo le risorse umane a loro disposizione. Il dialogo dipendente-responsabile non è verticale. La cultura aziendale in questa direzione è molto drastica: favorire uno scambio di idee equo, semplice e rompere le gerarchie.

In questo contesto, per l'intera durata del mio periodo di stage e dopo un primo momento di ambientamento ho beneficiato molto del clima aziendale. Infatti, non è mancato il libero confronto con il tutor aziendale che si è mostrato molto disponibile e aperto al mio spirito d'iniziativa. Lui mi ha supportato in ogni scelta che io ho motivato e ritenuto significativo per il beneficio del mio progetto.

Le idee sono una parte del processo di gestione dell'innovazione: la realtà è molto più complessa. IKS non possiede un effettivo processo di gestione a livello aziendale. Questo viene gestito da un gruppo di persone con competenze trasversali e a livelli organizzativi differenti.

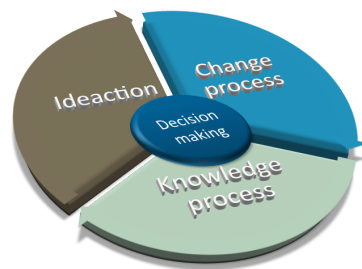


Figura 1.10: Il legame attivo tra innovazione e il processo del cambiamento e gestione della conoscenza. Immagine tratta da: <http://bit.ly/2qty3XD>.

Capitolo 2

Stage

Il presente capitolo è dedicato al rapporto dell'azienda con gli stage in collaborazione con l'Università nello specifico. A seguire presento l'ambito, le motivazioni, obiettivi aziendali e personali, e vincoli del mio progetto di stage.

Lo spirito dell'azienda è investire costantemente sui propri dipendenti e soddisfare i bisogni tecnologici del mercato con le giuste competenze. I dipendenti tecnici dell'azienda dedicano parte della propria giornata di lavoro in approfondimenti tecnologici e attività di laboratorio mirate a consolidare le proprie conoscenze. E' stato installato un ambiente virtuale basato sul prodotto open-source oVirt. Internamente è possibile sperimentare con nuove tecnologie, testare soluzioni di integrazione e mettere a disposizione degli stagisti un ambiente pronto all'uso. Oltre alla virtualizzazione è possibile sperimentare nei seguenti ambiti: storaging, networking e security. Alcuni temi comuni di approfondimento generale, invece, sono: Cloud, Machine Learning e Analytics.

Di recente, l'Azienda ha concluso una partnership strategica con AWS. L'accordo di collaborazione permetterà a IKS di portare molte delle proprie soluzioni sul Cloud pubblico e beneficiare delle sue peculiarità: elasticità, flessibilità di gestione di infrastrutture, usufruire di servizi specifici per esempio Elasticsearch a Service e molto altro ancora.

Negli ultimi anni, IKS ha sviluppato alcune soluzioni per il monitoraggio delle transazioni bancarie. Le fondamenta su cui queste soluzioni si basano sono le tecniche di Machine Learning e il monitoraggio di enormi moli di log. Inoltre, è in corso un progetto per il rilascio delle soluzioni di monitoraggio delle transazioni bancarie come servizio.

Ogni anno l'azienda partecipa a StageIT: evento dedicato a studenti universitari dei Corsi di Laurea in Scienze e Ingegneria Informatica, e aziende di Padova e Provincia. Infatti, quest'evento è un'opportunità per lo studente di mettersi in contatto con le realtà aziendali e per le aziende di conoscere i talent più da vicino. StageIT prevede per tradizione un concorso orientato alla scelta del progetto di stage più interessante e innovativo dell'edizione precedente dell'evento. In generale su un'insieme di stage vengono scelti un numero ristretto di finalisti i quali sono valutati in tempo reale dal pubblico presente alla manifestazione a seguito di una breve presentazione dei partecipanti. Il vincitore del concorso ottiene come premio un buono d'acquisto del valore di 500 Euro.

2.1 Il valore aggiunto di uno stagista

IKS è un partecipante attivo a StageIT e annualmente propone fino a 6 progetti di stage. Questi non sono verticali su un'unica tematica ma usualmente coinvolgono temi come:

- * Sviluppo di applicazioni basate su web, [Cloud](#), mobile o migrazione su [Cloud](#)/mobile di applicazioni tradizionali;
- * Progettazione di ambienti, metodologie e strumenti di sviluppo software.

Lo stagista è una risorsa importante per l'azienda. Esso viene visto come un portatore di novità. In principio è impiegato su progetti di sperimentazione. I quali hanno come obiettivo lo studio e l'analisi di fattibilità dell'integrazione delle soluzioni nell'offerta commerciale dell'azienda.

Per l'intera durata dello stage lo stagista si emerge in un ambiente di lavoro il più possibile vero simile alla realtà aziendale. Questo permette al tutor esterno di analizzare più da vicino il candidato in stage. E alla fine dello stage allo stagista può essere proposta un'offerta d'assunzione.

L'Azienda, grazie al contributo degli stagisti, si allinea con i temi di ricerca universitari e con le tendenze tecnologiche del momento sul mercato internazionale.

2.2 Alcuni temi di stage

2.2.1 AIOps e Machine Learning

Il progetto di stage tratta l'integrazione del Machine Learning con strumenti di Application Performance Monitoring. L'obiettivo dello stage è sperimentare integrando diverse soluzioni in questo ambito e studiarne il prodotto finale. Una conseguenza critica di questo progetto è lo sviluppo di un pensiero critico per affrontare le più difficili sfide del monitoraggio di applicazioni e infrastrutture.

2.2.2 DevOps Automazione

L'automazione è fondamento di ogni realtà aziendale contemporanea. Infatti, il numero di macchine da gestire spesso non è piccolo. Per semplificare i compiti di gestione si devono utilizzare strumenti di configurazione e di automazione. Queste tecnologie permettono di automatizzare tutte le operazioni manuali che un sistemista spesso compie durante le attività di manutenzione giornaliera. L'obiettivo di questo progetto è l'integrazione di alcuni strumenti che agevolano il [Patching](#) dei server e sperimentare con nuove tecnologie del settore.

2.2.3 Sviluppo moduli evolutivi in ambito antifrode

IKS ha grande esperienza in ambito della sicurezza informatica bancaria. Come prodotto risultate di questa esperienza è SMASH. L'obiettivo dello stage è estendere il prodotto con qualche funzionalità di monitoraggio di azioni sospette. Oltre allo sviluppo di moduli evolutivi lo stagista ha la possibilità di apprendere delle competenze forti nell'ambito della sicurezza informatica.

2.3 Il progetto proposto

2.3.1 Motivazioni

Sempre più comune è lo scenario di applicare internamente a una realtà aziendale l'approccio DevOps. In ambienti tradizionali il gruppo di persone che produce un prodotto software è diverso dal gruppo di persone che lo rilascia in produzione. Nel caso di aggiornamenti, per esempio correzione errori oppure nuove funzionalità, il tempo di rilascio è lunghissimo. L'approccio DevOps promuove l'unione dei due gruppi in un gruppo unico di lavoro che diventa responsabile sia dello sviluppo che del rilascio di un prodotto software. Il seguente approccio riduce il tempo di rilascio sul mercato del prodotto e/o funzionalità nuove per prodotti esistenti.

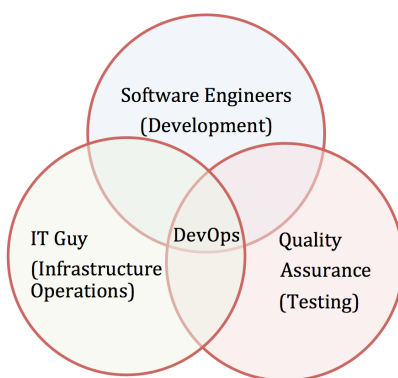


Figura 2.1: Visione d'insieme dell'approccio DevOps. Immagine tratta da: <http://bit.ly/2seXw3o>.

E' buona norma l'utilizzo di strumenti specifici che abilitano l'automazione in quanto le operazioni manuali sono complesse, lunghe e facilmente inclini a errori. La diffusione e l'impiego di questi strumenti nelle realtà aziendali hanno dato vita al concetto di *Infrastructure as Code*. Questo è un approccio orientato alla codifica della definizione di un'infrastruttura che si basa su un metodo disciplinato, quantificabile e sistematico. I vantaggi dell'infrastruttura come codice è la ripetibilità, portabilità dell'ambiente e la estensibilità dell'infrastruttura.

Nell'approccio DevOps non si considera unicamente l'infrastruttura, è importante prendere in causa anche le caratteristiche architetturali del prodotto software.

Assumendo i vincoli dell'architettura a monolite, un approccio sempre più comune e diffuso è scomporre in unità funzionali più piccole un prodotto complesso e gestire ogni funzionalità singolarmente. Lo strumento a supporto per la gestione di queste unità funzionali è la containerizzazione.

La tecnologia a container non è nuova. Un esempio di containerizzazione è il caso di BSD e la system call di chroot che permetteva negli anni '70 di cambiare la directory root di un processo e dei suoi relativi processi figli. In questo modo a molteplici prodotti software si permetteva di condividere lo stesso calcolatore ed evitare interferenza dell'uno sull'altro. Anche dal punto di vista della sicurezza, nel

caso in cui qualcuno comprometteva il software, chroot agendo come un involucro protettivo limitava gli accessi dall'esterno verso l'interno del server. Successivamente diversi tecniche per la containerizzazione sono state proposte. La più consolidata e riconosciuta come standard è la soluzione di Docker.

La containerizzazione confeziona mettendo insieme le dipendenze e il prodotto software in un'unica confezione. Segue che la sua installazione e successiva messa in esecuzione non richiede l'installazione locale di alcun altro software e tutti i container condividono lo stesso sistema operativo. Questo permette di semplificare la gestione di un servizio.

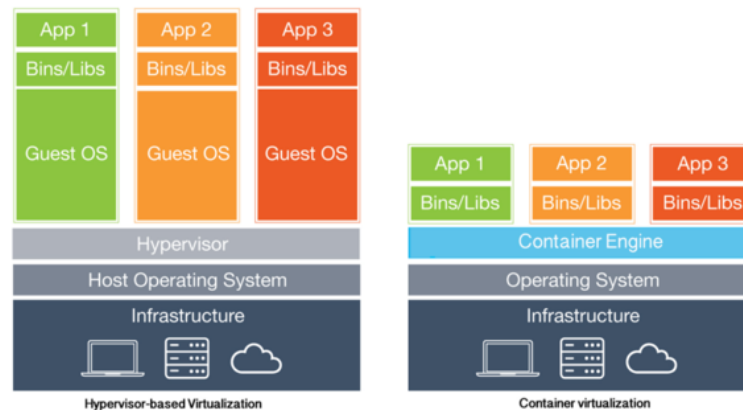


Figura 2.2: Vista a confronto del metodo classico di virtualizzazione basato su macchine virtuali con il nuovo metodo di virtualizzazione leggera basato su container. Immagine tratta da: <http://dockr.ly/23gH9Ad>.

Nonostante i numerosi benefici di Docker, la soluzione è limitata a un'unica macchina fisica. Per questo motivo in caso di requisiti come distribuzione, alta affidabilità e comunicazione inter container su più macchine è necessario integrare lo strumento con tool esterni. Un esempio è Kubernetes.

Kubernetes è una piattaforma open source sviluppata sopra docker da Google. Questo **Framework** diventa sempre più utilizzato per l'implementazione di microservizi.

I microservizi sono uno stile architetturale alla base delle applicazioni orientate al **Cloud**. Una simile applicazione è costituita da un insieme di servizi indipendenti che comunicano tra di loro attraverso la rete utilizzando il protocollo HTTP.

Le caratteristiche più evidenti dei microservizi secondo Martin Fowler sono:

- * *Componentizzazione*: abilità di sostituibilità di parti di un sistema con altri;
- * *Focus sul business* : lo sviluppo delle componenti è concentrato attorno alle necessità del business piuttosto che alla tecnologia;
- * *Rappresentando un sistema attraverso un grafo*, dove i nodi sono unità funzionali e le interconnessioni sono linee di rete per la comunicazione, segue che i nodi sono unità intelligenti: i portatori di valore aggiunto;

- * *Gestione decentralizzata dei dati*: ogni microservizio è responsabile per il proprio database;
- * *Predisposizione per l'automazione e il Continuous Delivery*.

Esistono diversi approcci per implementare sistemi complessi caratterizzati da uno stile architetturale a microservizi. Uno di questi è quello che ho precedentemente esposto: confezionare le applicazioni come un insieme di container e orchestrarli tramite uno strumento di orchestrazione, nello specifico Kubernetes.

2.3.2 Obiettivi aziendali

IKS ha realizzato una soluzione di Executive e Malware Dashboard basata sullo stack applicativo: Elasticsearch, Logstash e Kibana. La soluzione, progettata secondo uno stile architetturale SOA (Service Oriented Architecture), è complessa da gestire negli scenari di scalabilità, configurazione e riproducibilità degli ambienti.

In seguito a un'analisi di fattibilità è emerso che un'alternativa valida architetturale della soluzione sono i microservizi. Invece, lato tecnologico sono state individuate le tecnologie di supporto all'implementazione: docker per la containerizzazione delle componenti e kubernetes per l'orchestrazione di container, la scalabilità orizzontale, il self-healing ed ecc.

A questo scopo l'azienda ha delineato i seguenti obiettivi principali per lo stage:

- * Definire le modalità di containerizzazione delle singole componenti dello stack applicativo e loro attuazione;
- * Definire le modalità di messa in esercizio delle componenti containerizzazione in un ambiente privo di orchestratore e loro attuazione;
- * Installazione e configurazione dell'orchestratore;
- * Definire le modalità di messa in esercizio delle componenti nell'ambiente con orchestratore.

In breve, con il presente progetto di stage l'azienda ha studiato e sperimentato una soluzione architetturale alternativa per le Executive e Malware Dashboard. Nello specifico la soluzione a microservizi garantisce la robustezza, l'alta affidabilità e prediligere la flessibilità di configurazione e riduce il legame con l'infrastruttura fisica.

2.3.3 Obiettivi personali

Come attività preliminare alla ricerca di un progetto di stage per la Laurea ho attuato uno studio individuale di mercato. Lo scopo era capire: tendenze tecnologiche, architetture e metodologiche. Se da un lato le mie ricerche hanno cercato di cogliere le novità del momento, dall'altro a livello personale queste erano mirate alla ricerca di un contesto in cui potermi applicare e maturare.

Con il presente progetto gli obiettivi personali erano:

- * Apprendere conoscenze e competenze in ambito:
 - Della virtualizzazione basata sulla tecnologia a container;
 - Dei sistemi distribuiti;
 - Dell'amministrazione di sistema Linux;

—

- * Acquisire esperienza d'amministrazione delle reti di calcolatori in ambito dei sistemi, nello specifico le reti definite in modo programmatico per le tecnologie orientate alla containerizzazione;
- * Acquisire esperienza nell'analisi, progettazione e implementazione di sistemi orientati ai microservizi;
- * Imparare a livello base la piattaforma Kubernetes e i principi del [Cloud](#).

2.4 Piano di lavoro

Il piano di lavoro è stato pianificato per un totale di 300 ore complessive. Il contenuto del piano è stato presentato in un documento di cui una copia è stata consegnata all'Ufficio degli Stage presso l'Ateneo dell'Università di Padova, una seconda copia è stata consegnata al tutor interno e l'ultima copia controfirmata dall'ufficio stage dell'Università è stata consegnata all'azienda. Il piano è stato strutturato in tre fasi il cui contenuto presento di seguito:

- * Fase 1 - Formazione (56 ore)
 - Docker: la tecnologia per la containerizzazione;
 - Kubernetes: la tecnologia per l'orchestrazione;
 - ELK: lo stack applicativo;
 - Verifiche delle competenze acquisite;
- * Fase 2 - Analisi e progettazione (56 ore)
 - Analisi delle funzionalità della soluzione non containerizzata di dashboard;
 - Analisi delle modalità di containerizzazione delle componenti;
 - Analisi delle modalità di [Deployment](#);
 - Progettazione delle modalità di verifica della non regressione;
 - Progettazione architetturale della soluzione;
 - Progettazione della modalità di [Deployment](#);
 - Documentazione;
- * Fase 3 - Implementazione (188 ore)
 - Installazione e configurazione dell'orchestratore;
 - Implementazione della soluzione in un contesto con e senza orchestratore;
 - Verifica di non regressione;
 - Documentazione.

2.5 Vincoli

2.5.1 Vincoli temporali

Lo stage è durato 8 settimane per un complessivo di 310 ore di lavoro. Ho lavorato a tempo pieno con il seguente orario: 9.00-18.00. Con la pausa pranzo di 1 ora dalle 12.30 alle 13.30. Come stabilito nel PdL (Piano di Lavoro) le attività sono state strutturate in tre fasi. Ogni fase ha coinvolto attività mirate al raggiungimento di specifici obiettivi. Per maggior dettaglio sul contenuto del PdL riferire la [sezione Piano di Lavoro](#).

2.5.2 Vincoli tecnologici

Fin dal primo giorno di lavoro l'azienda mi ha fornito un portatile dedicato per l'intero periodo di stage. Inoltre, mi è stato vietato di collegare alla rete aziendale qualsiasi dispositivo personale. Inoltre, il portatile di lavoro non poteva essere portato a casa. Per comunicare internamente sono stati utilizzati strumenti di messaggistica istantanea come Skype e la posta elettronica.

Oltre a questo vincolo, a livello tecnologico sono state fissate le seguenti tecnologie:

- * Come sistema operativo delle macchine virtuali è stato fissato l'utilizzo di CentOS7, la versione open source di RHEL7 di Red Hat;
- * Per la containerizzazione è stato fissato l'utilizzo di Docker;
- * Per l'orchestrazione di container è stato fissato l'utilizzo di Kubernetes;
- * Per l'analisi di dati è stato fissato l'utilizzo di Elasticsearch;
- * Per l'ingestione di dati è stato fissato l'utilizzo di Logstash;
- * Per la visualizzazione dei dati è stato fissato l'utilizzo di Kibana.

Inizialmente sono state fissate anche le rispettive versioni delle componenti sopra citate. Tuttavia, nel corso dello stage ho realizzato che bloccare l'evoluzione di un'infrastruttura può comportare qualche problema nel futuro. A questo scopo ho predisposto un ambiente tollerante agli aggiornamenti e che si auto aggiorna. Durante la personalizzazione dell'ambiente mi sono ispirato al principio *self driven infrastructure* di CoreOS. In questo modo è possibile beneficiare di un costante aggiornamento delle componenti a partire dal bottom per arrivare al top le applicazioni che eseguono sull'infrastruttura.

Capitolo 3

Lo svolgimento dello stage

TODO: Aggiungere sintesi al capitolo

3.1 Metodo di lavoro

3.2 Attività di formazione

3.3 Analisi dei requisiti

3.3.1 Classificazione dei requisiti

3.3.2 Requisiti

3.3.2.1 Funzionali

3.3.2.2 Non funzionali

3.4 Progettazione e realizzazione

3.4.1 Scelte progettuali

3.4.2 Visione architetturale a microservizi

3.4.3 Codifica

3.4.4 Test

3.4.4.1 Test di carico

3.4.4.2 Test di durata

Capitolo 4

Valutazioni retrospettive

TODO: Aggiungere sintesi al capitolo

4.1 Obiettivi raggiunti

4.2 Problematiche riscontrate

4.3 Bilancio formativo

4.3.1 Il prima

4.3.2 Il dopo

4.4 Valutazione critica del Corso di Laurea

Glossario

Agile Metodo per lo sviluppo del software che coinvolge quanto più possibile il committente, ottenendo in tal modo una elevata reattività alle sue richieste. . 1, 23

Cloud Paradigma di erogazione di risorse informatiche, come l'archiviazione, l'elaborazione o la trasmissione di dati, caratterizzato dalla disponibilità on demand attraverso Internet a partire da un insieme di risorse preesistenti e configurabili. . 12, 14, 16, 23

Deployment Consegna o rilascio al cliente, con relativa installazione e messa in funzione o esercizio, di una applicazione o di un sistema software tipicamente all'interno di un sistema informatico aziendale. . 16, 23

Framework Architettura logica di supporto su cui un software può essere progettato e realizzato, spesso facilitandone lo sviluppo da parte del programmatore. La sua funzione è quella di creare una infrastruttura generale, lasciando al programmatore il contenuto vero e proprio dell'applicazione. . vii, 4, 14, 23

ICT insieme di metodi e tecnologie che implementano i sistemi di trasmissione, ricezione e elaborazione di informazioni.. 25

IT utilizzo di qualsiasi tecnologia di calcolo per offrire servizio di memorizzazione, reti per creare, processare, memorizzare e mettere in sicurezza ogni forma immaginabile di dato elettronico. . 25

Patching Applicare una patch, porzione di software progettata per aggiornare o migliorare un programma. Una patch permette di risolvere vulnerabilità di sicurezza e altri BugFix di un applicativo sviluppato. . 12, 23

Private equity Da definire . 1, 23

Acronimi

ICT [Information and Communication Technology](#). 1

IT [Information Technology](#). 1

Riferimenti

Bibliografia

- Baier, Jonathan. *Getting Started With Kubernetes*. PACKT, 2015.
- Clinton Gormley, Zachary Tong. *Elasticsearch: The Definitive Guide*. O'Reilly Media, 2015.
- Rafal Kuć, Marek Rogoziński. *Elasticsearch Server*. PACKT, 2016.
- Turnbull, James. *The Docker Book: Containerization is the new virtualization*. James Turnbull, 2016.
- Vohra, Deepak. *Kubernetes Microservices With Docker*. Apress, 2016.

Sitografia

- Documentazione Docker*. URL: <https://docs.docker.com/>.
- Documentazione Elasticsearch*. URL: <https://www.elastic.co/guide/en/elasticsearch/reference/2.4/index.html>.
- Documentazione Kibana*. URL: <https://www.elastic.co/guide/en/kibana/4.6/index.html>.
- Documentazione Kubernetes*. URL: <https://kubernetes.io/docs/home/>.
- Documentazione Linux*. URL: <http://www.tldp.org/>.
- Documentazione Logstash*. URL: <https://www.elastic.co/guide/en/logstash/2.3/index.html>.
- Documentazione Nginx*. URL: <https://nginx.org/en/docs/>.
- Martin Fowler: Micorservices*. URL: <https://martinfowler.com/articles/microservices.html>.
- Pattern architetturale a microservizi*. URL: <http://microservices.io/index.html>.
- Wikipedia: Cloud computing*. URL: https://en.wikipedia.org/wiki/Cloud_computing.
- Wikipedia: Microservices*. URL: <https://en.wikipedia.org/wiki/Microservices>.