

**PROCEDURA APERTA PER LA CONCLUSIONE DI UN ACCORDO QUADRO AVENTE AD OGGETTO
L’AFFIDAMENTO DI SERVIZI APPLICATIVI PER L’AREA ISTITUZIONALE DI INAIL – ED.3 – ID
2748**

APPENDICE 3 AL CAPITOLATO TECNICO

**LINEE GUIDA APPLICATIVE, PIATTAFORMA DI DELIVERY, CERTIFICAZIONE E MONITORAGGIO
DEI SERVIZI**

INDICE

1. ARCHITETTURE APPLICATIVE	3
1.1 THREE-TIER	3
1.2 MICROSERVIZI	3
1.3 SINGLE PAGE APPLICATION	4
1.4 CLOUD IBRIDO	4
2. DESCRIZIONE DELLE COMPONENTI SOFTWARE	6
3. LA PIATTAFORMA DI DELIVERY PER GOVERNARE I RILASCI	6
3.1 DESCRIZIONE DELLA PIATTAFORMA	7
3.2 MODALITÀ DI UTILIZZO DELLA PIATTAFORMA DI DELIVERY	8
3.3 IL CATALOGO DI BLUEPRINT GESTITE AD OGGI	9
3.4 TIPOLOGIE DI SOFTWARE ANCORA NON GESTITE IN PIATTAFORMA	9
4 CERTIFICAZIONE E MONITORAGGIO	10
4.1 CERTIFICAZIONE DEL SOFTWARE	10
4.2 TOOL UTILIZZATI ATTUALMENTE PER L'ESECUZIONE DEI TEST NON FUNZIONALI	13
4.3 TEST DI SERVIZIO	16
4.4 MONITORAGGIO – SERVICE CONTROL ROOM	17
4.5 STRUMENTO DI SERVICE MONITORING	18
4.6 STRUMENTI DI IT MONITORING	19

1. Architetture applicative

Le architetture applicative presenti in Inail si possono ricondurre alle seguenti tipologie: three-tier, microservizi, Single Page Application e soluzioni applicative in modalità Cloud ibrido.

1.1 Three-tier

L'architettura three-tier prevede una distribuzione dei diversi layer architetturali su una serie di componenti infrastrutturali:

- Componente Front-End → Oracle WebLogic Server
- Componente Back-End → JBoss EAP
- Componente Dati → Oracle, MS SQL Server, DB2 LUV, MongoDB

La sicurezza è gestita su WebLogic mediante l'utilizzo dello standard JAAS (Java Authentication e Authorization Services).

Nel perimetro delle applicazioni three-tier ricade anche l'utilizzo della SOA Suite di Oracle e dei prodotti di workflow management Oracle BPEL Process Manager e IBM Business Process Manager.

1.2 Microservizi

Negli ultimi anni si è consolidata in INAIL l'adozione delle architetture a microservizi. Tutti i nuovi sviluppi devono essere orientati a questo modello architetturale che porta a dare un rilievo sempre maggiore alle API come prodotto in sé.

Per poter abilitare il rilascio di soluzioni a microservizi, l'Istituto si è dotato di opportune tecnologie infrastrutturali:

- **API Gateway** - utilizzato per l'esposizione delle API Rest
- **OpenShift** – piattaforma di runtime per i microservizi
- **AMQ Broker** – soluzione per l'implementazione della comunicazione asincrona

Per quanto concerne la componente dati i prodotti di riferimento sono Oracle, MS SQL Server, DB2 LUV, MongoDB.

Per l'implementazione dei microservizi l'attuale framework di sviluppo di riferimento è SpringBoot.

La presenza di OpenShift fa sì che ogni microservizio per essere rilasciato sotto forma di container Docker.

1.3 Single Page Application

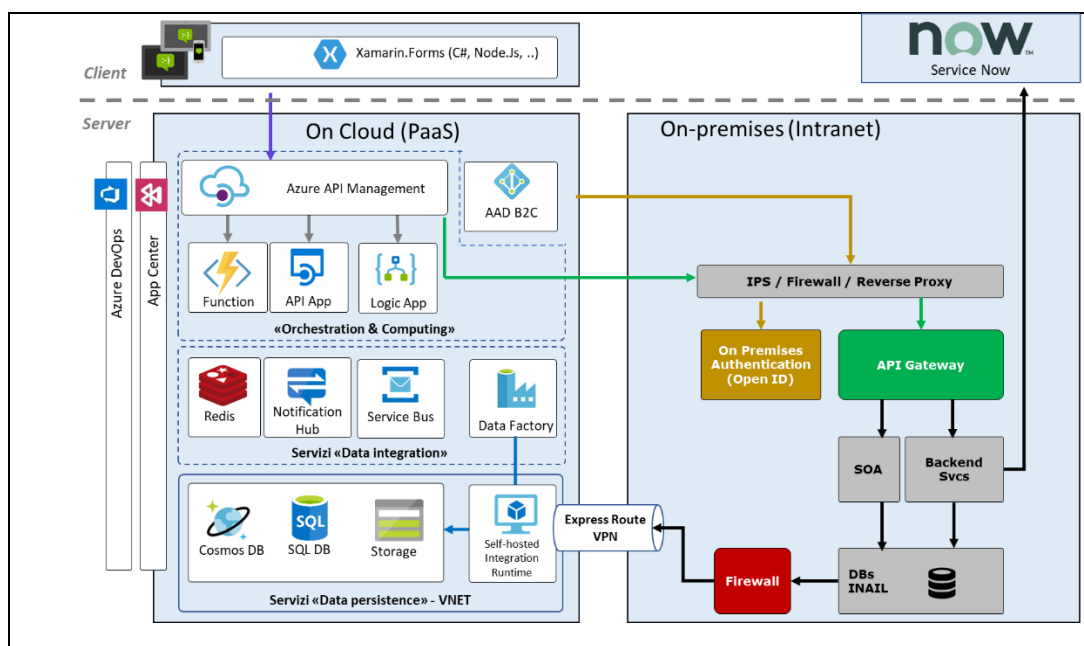
L'Istituto da avviato da tempo l'adozione di tecnologie per lo sviluppo di applicazioni di FrontEnd basate sul paradigma delle Single Page Application. Sono state realizzate sia come layer di Presentation per le applicazioni Three-tier, in questo caso il componente di FrontEnd rilasciato su WebLogic espone delle API Rest, che nel più consono caso d'uso di layer di Presentation per le applicazioni a microservizi, invocando le API esposte dall'API Gateway.

INAIL ha adottato Angular quale framework di riferimento per lo sviluppo delle SPA ed ha realizzato anche una libreria di componenti Angular che implementa il WebKit dell'Istituto.

1.4 Cloud ibrido

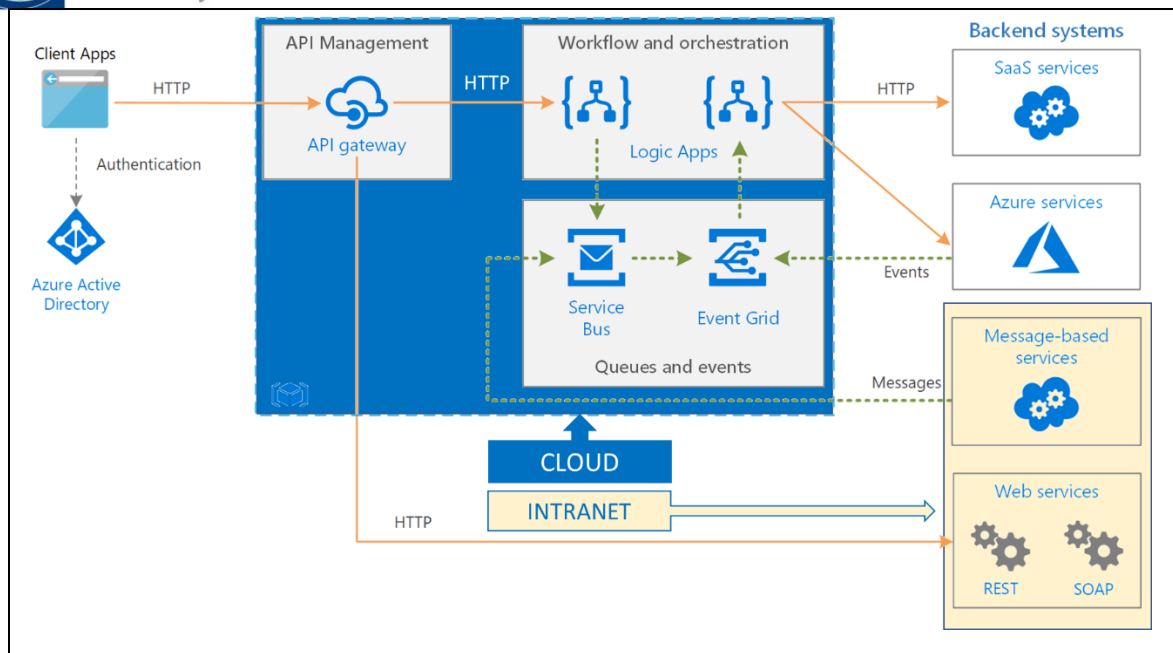
Sono applicazioni che sono distribuite in parte sul Cloud Pubblico ed in parte on premise. Il Cloud Pubblico è attualmente attestato su Microsoft Azure ma sono in corso ulteriori estensioni su altri Cloud come IBM e Oracle.

Di seguito lo schema architetturale d'insieme, che nei paragrafi successivi verrà dettagliato:



Le scelte architetturelle esposte nel presente documento nascono come adattamento alle specificità di INAIL della **Azure Reference Architecture**.

Il disegno di sintesi del suddetto riferimento architetturelle è il seguente:



Un generico workload basato su servizi Azure (leggasi: servizi Azure di tipo PaaS), nonché integrato nell’ecosistema dei servizi INAIL, se analizzato a livelli di profondità successivi può così scomporsi:

- Servizi **PaaS** in Public Cloud;
- Servizi di **Back End intranet**, intermediati verso i Consumer dall’API Gateway.

Per quanto riguarda i servizi di Back-end intranet si distinguono due tipi di end-point:

- **REST API** esposte da API Gateway
- **SOAP** Web Service esposti o meno come REST API su API Gateway

La più stretta integrazione tra workload Hybrid Cloud e l’ecosistema INAIL avviene negli ambiti di Autenticazione e Autorizzazione. In DCOD tali ambiti sono in carico al software **CA SiteMinder** e all’ecosistema **IAM (Identity Access Management)** su di esso incentrato.

L’integrazione dell’Hybrid Cloud con l’IAM dell’Istituto si realizza mediante l’utilizzo dei seguenti servizi Azure:

- **Azure Active Directory (Azure AD)**
- **Azure AD B2C (Business to Consumer)**

Azure AD è il repository in Cloud delle utenze del personale INAIL. Già attivo, Azure AD è alla base del servizio Office 365. Le User ID sono le medesime che esistono sul dominio “INAILUTENTI” della foresta Windows Active Directory interna.

Azure AD B2C è la componente di Azure AD che consente l’integrazione con altri Identity Provider (IdP) sugli standard OAuth2 e OpenID Connect. È questa la componente che abilita l’integrazione tra Azure e il contesto autorizzativo INAIL.

2. Descrizione delle componenti software

Attualmente la descrizione di dettaglio delle architetture, in riferimento a uno specifico prodotto da rilasciare, è delegata a documenti cartacei associati al rilascio: le attività necessarie alla creazione dell'infrastruttura per il rilascio e al deploy dei componenti su tale infrastruttura hanno come riferimento tale documentazione, sono manuali e sono governate rispettivamente da richieste su Service Now e su Rational Team Concert (RTC), richieste che il team Dev (sviluppo) rivolge a vari gruppi Ops (operations).

Su questa realtà consolidata sono subentrati nel tempo e sono già operativi alcuni fatti nuovi:

- l'introduzione in INAIL di tecnologie nativamente orientate al mondo del cloud: OpenShift, in primo luogo, che ha di fatto soppiantato il ruolo centrale dell'application server;
- la presenza sempre più rilevante di sviluppi che utilizzano servizi (IaaS, Paas e SaaS) resi disponibili dai vari cloud provider pubblici;
- la tendenza a proporre l'utilizzo di architetture applicative a microservizi per la componente di back end;
- la volontà di dismettere progressivamente RTC;
- le API che acquisiscono sempre maggior rilievo come prodotto software autonomo.

Tale quadro ha notevolmente complicato il processo di rilascio del software e ha indotto INAIL a standardizzare il processo e le attività legate al rilascio del software, introducendo un servizio (una piattaforma di delivery) che fosse in grado di comprendere anche le fasi di descrizione formale delle componenti, al fine di orchestrarne/governarne il rilascio, introducendo in aggiunta pratiche di deploy automatico: il servizio DevOps o Piattaforma di delivery.

Esiste un mondo applicativo che continuerà per qualche tempo ad utilizzare gli usuali strumenti che governano la gestione manuale dei rilasci. Tuttavia la Piattaforma di Delivery, gradualmente, prenderà in carico tutto il software.

Tutti i nuovi sviluppi però, fin da subito, dovranno nascere con l'idea che il software da essi prodotto dovrà essere descritto e gestito attraverso la Piattaforma di Delivery descritta nel paragrafo successivo.

3. La piattaforma di delivery per governare i rilasci

La Piattaforma di delivery di INAIL è uno strumento attraverso il quale è possibile gestire l'evoluzione di prodotti software a catalogo INAIL e a catalogo delle PA a cui INAIL eroga servizi (es. Ministero della Salute).

In una realtà pubblica come quella di INAIL, le attività di sviluppo sono delegate a una molteplicità di fornitori ai quali, periodicamente, subentrano altri fornitori in virtù dell'esito di gare d'appalto. A questa pluralità di soggetti che sviluppano software per conto di INAIL daremo il nome di Dev (sviluppo).

Esso consente di:

- Governare l'inserimento di un nuovo prodotto in un catalogo
- Governare l'evoluzione di un prodotto già esistente in un catalogo

Il cuore della Piattaforma è il DevOps propriamente detto, ovvero la possibilità di gestire il rilascio di componenti di prodotto per mezzo di pipeline automatiche; la piattaforma è per l'Amministrazione uno strumento di governo di tutti i rilasci e non solo una semplice facility per lo Sviluppo (Dev).

Tale **piattaforma di delivery** consente a Dev di descrivere il software da rilasciare secondo formalismi che rendono possibile la gestione automatizzata dei rilasci e dei suoi componenti su infrastrutture eterogenee, e abilitano il DevOps. Di fatto la piattaforma consente una gestione strutturata dei rilasci, è in grado di adeguarsi a nuove future tecnologie introdotte dallo sviluppo e a possibilità di realizzare potenziamenti in termini di risorse di elaborazione.

Gli elementi di base con cui descrivere in piattaforma il software da rilasciare sono le **blueprint**, ossia file testuali in formato YAML con cui descrivere classi di prodotto istanziabili tramite la piattaforma di delivery.

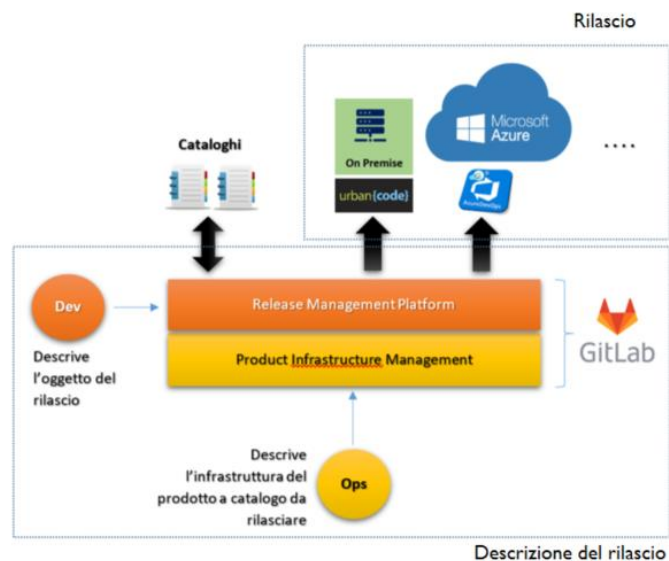
Dopo aver associato il software da rilasciare a una blueprint, il Dev si limita a dettagliarne i contenuti in piattaforma e ad associare i sorgenti che, accanto a tale descrizione, rendono possibile il DevOps.

La piattaforma di delivery consente poi di associare una release al software da rilasciare, si integra con i processi di accettazione, interfacciandosi con i sistemi di test management, e gestendo i processi autorizzativi interni associati al rilascio. In sintesi, la piattaforma lavora gestendo un catalogo di blueprint e alimenta un catalogo di prodotti software generati e rilasciati a partire dalle blueprint definite a catalogo.

3.1 Descrizione della piattaforma

La piattaforma è un **insieme di tool custom e di prodotti di mercato** che offrono a Dev e a Ops la possibilità di:

- **descrivere** il software da rilasciare nei suoi componenti e i run time environment che ospiteranno tali componenti
- **rilasciare** tali componenti sui run time environment in cui essi saranno eseguiti (che potranno anche essere eterogenei)



I tool custom RMP (Release Management Platform) e PIM (Product Infrastructure Management) servono al Dev e a Ops per descrivere (su GitLAB) il prodotto da rilasciare e l'infrastruttura da associare al rilascio.

Su repository GitLAB, predisposto a valle della descrizione fatta su RMP, il Dev dovrà rilasciare i source e/o le configurazioni a partire dalle quali la piattaforma gestirà i rilasci sulle infrastrutture rese disponibili da OPS attraverso processi interni e descritte sempre da OPS su PIM.

Tali rilasci saranno gestiti utilizzando prodotti e strumenti messi a disposizione dalle infrastrutture on-premise e/o cloud utilizzate (in figura Azure e Azure DevOps sono soltanto un esempio).

3.2 Modalità di utilizzo della piattaforma di delivery

I delivery di ogni nuovo sviluppo dovranno essere gestiti utilizzando la piattaforma di delivery di INAIL. È in corso un processo di riconduzione anche del software esistente all'interno di tale piattaforma.

Prima di avviare ogni sviluppo è opportuno accertarsi che l'oggetto dello sviluppo sia descrivibile nella piattaforma di deliverable. Nel caso non lo fosse va attivato il processo interno che gestisce la definizione di nuove blueprint (o l'adeguamento di quelle esistenti) e in quella fase andranno definite le modalità con le quali andranno gestiti i rilasci dei componenti previsti dalle nuove blueprint (modalità che saranno automatiche, nella maggior parte dei casi).

3.3 Il catalogo di blueprint gestite ad oggi

Il catalogo delle blueprint attualmente gestite dalla piattaforma è il seguente:

Tipologia di prodotto	Classe di prodotto (blueprint)	Descrizione
Front end	Spa monolitica	L'Istituto da avviato da tempo l'adozione di tecnologie per lo sviluppo di applicazioni di FrontEnd basate sul paradigma delle Single Page Application che invocano API esposte dall'API Gateway. La blueprint oggi gestisce la tecnologia Angular
Headless	Servizi ocp	<p>Negli ultimi anni si è consolidata in INAIL l'adozione delle architetture a microservizi. Tutti i nuovi sviluppi devono essere orientati a questo modello architetturale che porta a dare un rilievo sempre maggiore alle API come prodotto in sé.</p> <p>Per poter abilitare il rilascio di soluzioni a microservizi, l'Istituto si è dotato di opportune tecnologie infrastrutturali:</p> <ul style="list-style-type: none"> • API Gateway - utilizzato per l'esposizione delle API Rest • OpenShift – piattaforma di runtime per i microservizi • AMQ Broker e AMQ Stream – soluzione per l'implementazione della comunicazione asincrona <p>La blueprint oggi gestisce le seguenti tecnologie: SpringBoot, Quarkus, NodeJS, Python per le componenti di logica applicativa, Oracle, MS SQL Server e MongoDB per le componenti dati.</p>
	Servizi ocp dati condivisi	Questa blueprint è simile alla precedente, con la differenza che la componente dati del software è condivisa tra tutti i servizi

Tale catalogo è in continua evoluzione.

3.4 Tipologie di software ancora non gestite in piattaforma

A questa categoria appartengono due tipologie di applicazioni:

- **le applicazioni three-tier che prevedono rilasci su application server**

Come già scritto, i nuovi sviluppi non utilizzeranno questa architettura, ma il rilascio di evolutive/correttive sui software esistenti saranno gestiti nella piattaforma di delivery

- **le applicazioni cloud**

È prevista una gestione futura dei rilasci di queste applicazioni utilizzando la piattaforma di delivery

4 CERTIFICAZIONE E MONITORAGGIO

4.1 Certificazione del Software

Le attività di “certificazione del software” hanno l’obiettivo di garantire, per tutte le applicazioni oggetto di certificazione, con l’esecuzione dei “test non funzionali”, i risultati attesi in termini di qualità statica del codice, accessibilità, sicurezza e prestazioni.

I principali obiettivi del processo di "certificazione del Software" sono i seguenti:

- verificare che i requisiti “non funzionali” siano soddisfatti prima del passaggio in produzione
- verificare l’aderenza agli standard UE di qualità, sicurezza ed accessibilità e previsti dall’Istituto.

Le attività svolte nell’ambito della certificazione del software sono:

- test di qualità statica del codice
- test di accessibilità
- test di sicurezza applicativa (SAST, DAST, SCA, PT)
- test prestazionali

Nel dettaglio sono effettuate le seguenti analisi/verifiche sul software sviluppato e/o modificato:

Qualità statica del codice: verifica di ottimizzazione applicativa e valutazione del codice sorgente di un sistema o di un suo componente basato sulla sua forma, sulla sua struttura, sul suo contenuto, le quali forniscono elementi utili al miglioramento di tutte le componenti software applicative che costituiscono il sistema. L’attività è svolta in automatico durante fase di deploy in ambiente di collaudo.

I test sono eseguiti sul codice sorgente oggetto di rilascio, tenendo conto dell’utilizzo di Standard Internazionali per la verifica dei Rischi di Qualità (ISO5055).

A valle dell’esecuzione dei test è prodotta la relazione dei Test statici di qualità che ne illustra i risultati.

Accessibilità: verifica della compliance rispetto alle Normative Vigenti e alle Linee Guida relative, con riferimento al Vademecum per la misurazione della qualità dei siti web delle PA e alla Legge 4/2004 (“Disposizioni per favorire l'accesso dei soggetti disabili agli strumenti informatici”) e in relazione alle nuove disposizioni emanate da AGID (vedi norma tecnica EN 301549). I test di accessibilità sono di tipo automatico e manuale; tuttavia, la valutazione di accessibilità di avvale della revisione manuale condotta dal tester di certificazione. Le WCAG di riferimento utilizzate per verificare l’accessibilità dell’applicazione web sono le 2.1/2.2. A valle dell’esecuzione dei test è prodotta e inviata la relazione dei Test di Accessibilità che ne illustra i risultati

Sicurezza applicativa: L’attività ha lo scopo di verificare la vulnerabilità dell’applicazione prima che il sw venga rilasciato in Produzione.

Le tipologie di TNF eseguiti sono:

- Penetration Test (PT) (eseguiti manualmente in certificazione)
- Verifica librerie 3° parti (SCA) (automatici – eseguiti in fase di build-time)
- Analisi statica del codice sorgente (SAST) (automatici – eseguiti in fase di deploy in ambiente di collaudo)
- Analisi dinamica del codice (DAST) (automatici – eseguiti in fase di post-deploy in ambiente di collaudo)

I test sono effettuati con metodologia OWASP e OSSTMM sulla base di criteri descritti in apposite linee guida di certificazione, sempre utilizzando gli Standard Internazionali per la verifica dei Rischi di Sicurezza (OWASP, CWE, STIG, CISQ, NIST).

A valle dell’esecuzione dei test è prodotta e condivisa con i GdS la relazione dei Test di Sicurezza che ne illustra i risultati.

Per la gestione delle attività di test, viene utilizzato un tool interno all’Istituto, che tiene traccia dei test condotti sulle applicazioni oggetto del processo di verifica di sicurezza, nonché delle vulnerabilità individuate e delle relative remediation, e delle varie fasi che vanno dalla pianificazione dei test alla correzione delle vulnerabilità individuate con i relativi ricicli.

Prestazioni: verifica delle performance, utile a fornire elementi per il miglioramento dei tempi di risposta delle singole applicazioni in ottica E2E, prima del passaggio in produzione. Le verifiche prestazionali sono eseguite tramite un prodotto on-prem interna alla DCOD INAIL per le applicazioni web che prevedono fino a 5.000 utenti simultanei ed uno in cloud per applicazioni che necessitano un accesso simultaneo superiore a 5.000 utenti.

I test sono effettuati utilizzando una metodologia basata sull'indice APDEX, il cui scopo è quello di convertire le misurazioni in informazioni sulla soddisfazione degli utenti, specificando un modo uniforme di analizzare e riportare il grado di soddisfazione delle prestazioni misurate da parte degli utenti, sempre rispettando i tempi minimi previsti dall'Istituto in base anche ai tempi estratti real-time in fase di monitoraggio in esercizio.

A valle dell'esecuzione dei test è prodotta e condivisa con i GdS la relazione dei Test Prestazionali che ne illustra i risultati.

Al fine di essere sempre in linea con le nuove norme UE e/o ISO, sarà cura dell'Istituto verificare ed attuare eventuali aggiornamenti delle versioni e/o dei prodotti usati oggi in fase di esecuzione dei test non funzionali (TNF), con nuove versioni o altri prodotti presenti sul mercato più innovativi e tecnologici.

I risultati ottenuti dalle suddette verifiche, sono riportati nei seguenti documenti, condivisi con i Gruppi di Sviluppo:

- **Documento di analisi del codice**, contenente gli indicatori di qualità utilizzati e i risultati dell'analisi statica del codice in ambiente di test;
- **Documento di analisi dell'accessibilità dell'applicazione**, contenente gli esiti per tutte le pagine web che compongono l'applicazione stessa, con evidenza delle eventuali violazioni della normativa vigente;
- **Condivisione delle vulnerabilità riscontrate su una dashboard specifica e profilata per applicazione/GdS, tale da garantire la privacy e la non divulgazione delle vulnerabilità trovate sia dai tool automatici che dai test manuali.** Ove possibile, verranno proposte azioni correttive per mitigare o risolvere le difformità.
- **Documento di analisi della prestazione dell'applicazione**, contenente gli esiti delle verifiche prestazionali, con particolare attenzione ai tempi di risposta rilevati in ambiente di test.

4.2 Tool utilizzati attualmente per l'esecuzione dei Test Non Funzionali

Qualità statica del codice: CAST (INAIL)

CAST= Prodotto con licenza per il controllo continuo della qualità del codice.

Sono strumenti che garantiscono un'ispezione continua del codice e mettono a disposizione migliaia di regole automatizzate finalizzate all'analisi statica del codice.

Il test di analisi statica del codice viene eseguito in modalità automatica in fase di build; l'esito del test è consultabile direttamente su entrambi i prodotti in base alla toolchain utilizzata. In caso di build ripetute, è previsto che i tool conservino la cronologia delle analisi effettuate.

Per il superamento dei test di qualità, si tiene conto del Quality Gate che è l'insieme delle metriche o golden rule prese in esame per definire il risultato dell'analisi della qualità statica di un artefatto approvate dall'Istituto con le relative soglie.

Per CAST, invece sono disponibili delle VM dedicate ad uso esclusivo dei GdS, sulla quale è possibile effettuare la scansione del proprio software in totale autonomia, secondo delle LG condivise dal GdL Certificazione di Qualità, prima che lo stesso venga ufficialmente rilasciato sulle toolchain dell'Istituto.

Accessibilità: Lighthouse, JAWS e PAC

Questi strumenti aiutano a individuare i problemi di accessibilità e offrono suggerimenti su come risolverli. Tuttavia, è importante ricordare che gli stessi prodotti non possono sostituire l'esperienza umana.

Attraverso l'utilizzo del prodotto **JAWS**, al fine di garantire la massima facilità d'uso ed accessibilità di un'applicazione, i tester di certificazione, consultando il manuale utente fornito dai GdS, effettuano una navigazione manuale delle funzionalità applicative presenti sul FE. Per i test di accessibilità vengono prese in considerazione 2 versioni distinte del prodotto JAWS:

- In caso di applicazioni di tipo intranet, la versione di JAWS utilizzata sarà quella presente nelle varie sedi INAIL presenti sul territorio nazionale.
- In caso di applicazioni di tipo internet, la versione di JAWS utilizzata sarà l'ultima disponibile sullo store del vendor (Freedom Scientific).

Infine, p" (sempre ultima versione disponibile sul sito del vendor). Con questo strumento i documenti PDF vengono analizzati secondo i criteri di accessibilità "PDF/UA" e "WCAG 2.1", di seguito descritti:

PDF/UA (Universal Accessibility): PDF/UA è uno standard sviluppato per garantire l'accessibilità dei documenti PDF alle persone con disabilità. Si concentra sull'accessibilità dei contenuti del documento PDF, come testo, immagini, tabelle e elementi multimediali. PDF/UA specifica requisiti tecnici e linee guida per creare documenti PDF accessibili, inclusi i requisiti per la struttura del documento, i metadati accessibili, l'ordine di lettura logico, la corretta etichettatura degli elementi, l'accessibilità delle immagini e altro ancora.

WCAG 2.1 (Web Content Accessibility Guidelines): Le WCAG 2.1 sono linee guida sviluppate dal World Wide Web Consortium (W3C) per garantire l'accessibilità dei contenuti web. Sebbene non siano specificamente focalizzate sui documenti PDF, le linee guida WCAG 2.1 possono essere applicate anche ai PDF pubblicati sul web. Le WCAG 2.1 definiscono i criteri di conformità che i contenuti web devono soddisfare per essere accessibili a una vasta gamma di utenti, inclusi quelli con disabilità. Le linee guida WCAG 2.1 coprono aspetti come la percezione dei contenuti, la navigabilità, l'accessibilità delle forme, l'accessibilità dei media e altro ancora.

Sicurezza applicativa:

- **Burp suite Professional** (Open source) Portswigger BurpSuite è un toolkit per l'esecuzione di penetration test e security assessment web. Viene installato e configurato come un proxy in modo da intercettare il traffico HTTP tra il client e il server. L'edizione Professional mette a disposizione del tester varie funzionalità che consentono l'esecuzione dei test sia in maniera manuale (funzionalità Repeater, Intercept, etc.) che automatica (funzionalità Intruder, Automatic Scan, etc.).
- **ScanFortify** (TNF DAST). Test automatici eseguiti in fase di post-deploy in ambiente di collaudo, il cui scopo è di verificare il comportamento del sw in fase di runtime

- **CAST** (TNF SAST/SCA). Prodotto integrato nella toolchain tradizionale e devops per garantire un alto grado di sicurezza delle applicazioni Inail e di velocizzare il loro rilascio in produzione. Secondo la filosofia DevOps e paradigma Shift-Left, la scansione del codice sorgente e delle librerie di 3° parti è presente in fase di build-time e deploy del sw in ambiente di collaudo. Questo permette di:

- Velocizzare i controlli di sicurezza senza gravare troppo sui gruppi di lavoro coinvolti
- Individuare in anticipo problemi di sicurezza per avere un minor numero di ricicli nei test in certificazione, ottenendo così un incremento della produttività del ciclo di sviluppo
- Migliorare la qualità complessiva del software prodotto, incentivando la scrittura di codice migliore dal punto di vista della sicurezza fin dalle prime fasi del processo

Come descritto per Qualità, per poter anticipare la scansione del proprio codice ed intervenire per tempo sulle possibili vulnerabilità presenti in fase di sviluppo, è possibile effettuare in autonomia delle scansioni preliminari su delle VM messe a disposizione dei GdS senza dover passare per forza attraverso un rilascio ufficiale sulle toolchain utilizzate in INAL per il rilascio del sw in produzione.

Prestazioni: RPT

Il tool RPT è usato per i test prestazionali in ambiente di certificazione quando il carico che deve essere simulato non è particolarmente elevato.

Una soglia indicativa è quella di 5000 VU.

Quando il carico è più elevato o in casi particolari (necessità di evitare il proxy dell'ambiente di certificazione) si eseguono i test in cloud.

In questo caso il tool utilizzato per preparare gli script è Loadrunner, mentre per l'esecuzione si usa Loadrunner in cloud.

Le caratteristiche principali di RPT in relazione all'uso fatto in INAIL consentono:

- la creazione di script di test registrando le azioni svolte dal tester.
- La cattura del traffico di rete che viene generato quando l'applicazione in esame interagisce con un server. Questo traffico di rete viene poi emulato su più utenti virtuali durante la riproduzione del test.
- di supportare i test di carico su un'ampia base di applicazioni come HTTP, SAP, Siebel, TCP Socket e Citrix.
- di generare report sulle prestazioni e sul throughput in tempo reale, consentendo di individuare i problemi di prestazioni in qualsiasi momento durante un test.
- Di arricchire gli script con codice custom sviluppato in Java.

Il tool si articola su due componenti:

1. il Performance Test Perspective (Workbench) che è il componente utilizzato per sviluppare gli script; lanciare il test e raccogliere e analizzare i risultati
2. l'Agent che è il componente utilizzato assieme al Workbench per aumentare il numero dei Virtual User da iniettare

L'utilizzatore (tester) accede alla macchina ove è installato il Workbench, sviluppa lo scenario di test, esegue lo scenario e raccoglie e analizza i risultati. Al momento dell'esecuzione del test, per generare il numero di Virtual User desiderati, deciderà se utilizzare solo il Workbench o anche gli Agent ad esso associati.

4.3 Test di servizio

Il test di servizio include la verifica degli impatti del singolo rilascio sia sull'intera applicazione/servizio di business (oggetto di rilascio) sia con tutte le interazioni esterne che la stessa applicazione ha con altre applicazioni/servizi di business.

Procedura aperta per la conclusione di un Accordo Quadro avente ad oggetto l'affidamento di servizi applicativi per l'Area Istituzionale di INAIL – Ed.3 – ID 2748

Appendice 3 al Capitolato tecnico – Linee guida architetture applicative, processo di rilascio, certificazione e monitoraggio

Classificazione: Consip Public

Saranno oggetto di valutazione sia gli impatti in relazione al servizio di business rivolto all'utente finale (in ottica E2E) sia gli impatti legati ad esigenze tecnologiche di natura architettuale, infrastrutturale, etc.

Si riportano a titolo esemplificativo alcune verifiche inerenti il test di servizio:

- Verifica che tutte le funzionalità/percorsi di navigazione riguardanti l'applicazione e/o servizio rivolto all'utente finale in ottica end to end rispettino gli standard di performance previsti, con l'obiettivo di garantire la corretta esercibilità del servizio di Business per assicurare la qualità finale percepita dall'utente;
- verifica che tutte le componenti tecnologiche che concorrono all'erogazione del servizio rivolto all'utente finale rispondano correttamente secondo gli standard architeturali e infrastrutturali previsti. Tale modalità permette di evidenziare tempestivamente criticità e punti di caduta di tipo applicativo e/o infrastrutturale che, in caso contrario, potrebbero emergere soltanto in esercizio una volta rilasciata ed installata l'applicazione. Lo scopo principale è quello di ottenere una consapevolezza dello stato prestazionale dell'applicazione in esame all'interno dell'intero ecosistema Inail integrato.

4.4 Monitoraggio – Service Control Room

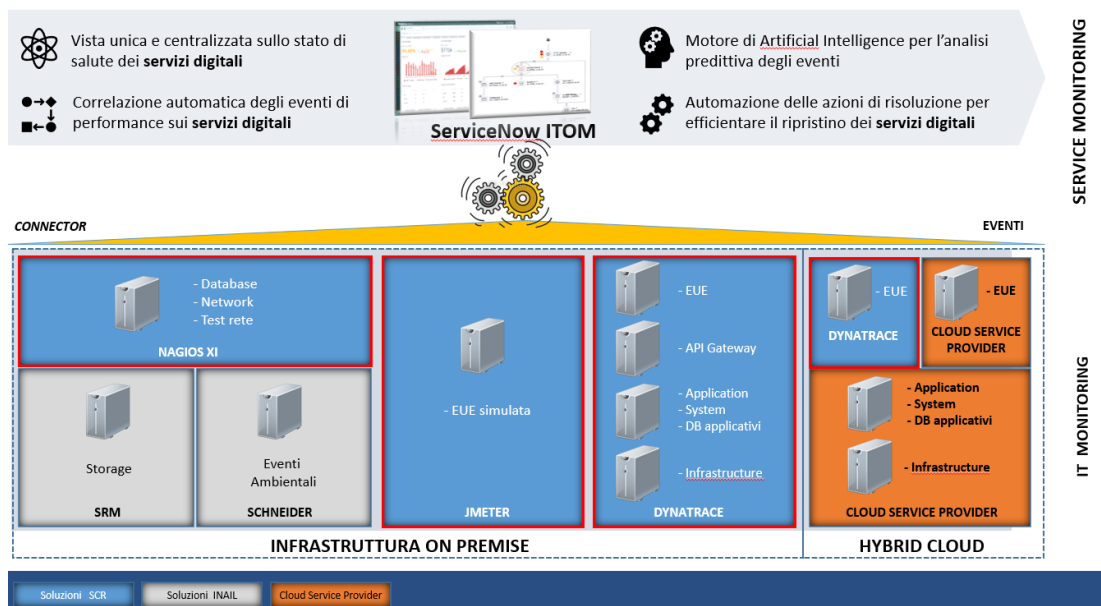
La Service Control Room è la funzione della DCOD (Direzione Centrale per l'Organizzazione Digitale) responsabile del monitoraggio tecnologico dei servizi digitali, in ottica utente finale erogati dall'Inail nei confronti dei propri interlocutori istituzionali.

La Service Control Room ha l'obiettivo di minimizzare i disservizi agli utenti finali (gestione proattiva) e massimizzare l'efficienza del processo di reazione ai malfunzionamenti, attraverso la gestione proattiva e predittiva degli eventi impattanti la disponibilità e performance dei servizi digitali erogati.

Tra gli obiettivi del monitoraggio finalizzati a minimizzare i disservizi agli utenti finali sui servizi di Business erogati dall'istituto, la SCR ha un ruolo chiave anche nell'implementazione proattiva della predisposizione del monitoraggio sull'insieme dei servizi in fase di sviluppo da parte degli uffici competenti. A tal fine il fornitore dovrà garantire la condivisione con la Service Control Room sin dalle fasi iniziali dei progetti delle specifiche, della documentazione e pianificazione delle attività, così come previsto dal modello processi interni adottati dall'Istituto.

L'architettura e soluzione complessiva della Service Control Room è composta da differenti strumenti di monitoraggio, specializzati su ambiti verticali che, opportunamente configurati in termini di regole di correlazione degli eventi sulle differenti componenti dei modelli di servizio, forniscono una vista unica ed integrata dello stato di salute di un servizio.

Di seguito si riporta il disegno architetturale della soluzione complessiva di monitoraggio adottata in service Control Room:



L'architettura e gli strumenti utilizzati attualmente dalla Service Control Room sono descritti nei successivi paragrafi.

4.5 Strumento di Service Monitoring

ServiceNow ITOM (SaaS platform) è lo strumento adottato per il service monitoring. La soluzione permette l'aggregazione e correlazione degli eventi di monitoraggio in ottica servizio di business end to end. Le caratteristiche funzionali della soluzione sono:

- Correlazione automatica degli eventi di disponibilità dei servizi

- Motore di Artificial Intelligence e predictive analysis tramite la creazione di baseline dinamiche sulle performance dei servizi
- IT Remediation integrata per la risoluzione automatizzata dei malfunzionamenti IT tramite workflow
- Utilizzo Knowledge Base Unica ed integrata all'interno dell'ecosistema ServiceNow
- Vista mobile e app native

Le caratteristiche tecnologiche della soluzione sono:

- Integrazione nativa con ServiceNow ITSM INAIL
- Integrazione nativa con ServiceNow CMDB INAIL
- Integrazione tramite connettore standard con Dynatrace
- Integrazione tramite connettore standard con Nagios XI
- Integrazione tramite WS Standard con MS AppInsight
- Semplicità di integrazione verso Tool esterni

Lo strumento, integrato insieme ai tool verticali di monitoraggio facenti parte il modello SCR (Dynatrace, Nagios XI e Jmeter) e con gli altri strumenti esterni (es: MS AppInsight) permette di avere una vista unica e centralizzata circa stato di salute dei servizi oggetto di monitoraggio al fine di garantire piena visibilità del servizio finale erogato e percepito dagli utenti

4.6 Strumenti di IT Monitoring

Gli strumenti verticali di monitoraggio adottati dalla Service Control Room sono i seguenti:

- Dynatrace
- Nagios XI
- Apache Jmeter

Dynatrace, leader di mercato tra le soluzioni di APM monitoring, è lo strumento che permette di avere una visione end to end dei servizi in grado di attraversare tutti gli ambiti tecnologici che concorrono all'erogazione del servizio finale. Le caratteristiche della soluzione sono le seguenti:

- **IM – Infrastructure Monitoring** : Consente il monitoraggio delle risorse (CPU, RAM, Spazio Disco, Traffico Rete) e lo stato dei processi di macchine virtuali, host fisici e

di Containers, oltre che i principali dati sul network, fornendo metriche infrastrutturali integrate con le applicative (Dynatrace Cluster)

- **APM – Application Monitoring:** Fornisce la ricostruzione automatica del flusso applicativo correlando tutte le componenti attraversate, fornendo dati di performance e monitoraggio applicativo, dal Client al DB, velocizzando le attività di analisi e risoluzione degli incidenti
- **Monitoraggio completo della User Experience, ovvero di tutte le azioni che gli utenti compiono durante la navigazione,** creando da queste delle metriche di business e permette di catturare l'intera sessione degli utenti e riprodurla in video; **attraverso le Synthetic Transactions, consente** inoltre di **eseguire navigazioni simulate da rete esterna**, da diverse location, monitorando il funzionamento dei servizi di business su cui potranno essere create delle dashboard apposite
- **DEM - Digital Experience Monitoring:** Monitoraggio completo della User Experience, ovvero di tutte le azioni che gli utenti compiono durante la navigazione, creando da queste delle metriche di business e permette di catturare l'intera sessione degli utenti e riprodurla in video; attraverso le Synthetic Transactions, consente inoltre di eseguire navigazioni simulate da rete esterna, da diverse location, monitorando il funzionamento dei servizi di business
- **AIOps – Artificial Intelligence:** Dynatrace One Agent fornisce un motore di Intelligenza Artificiale di tipo deterministico chiamato “Davis” che elabora e correla costantemente tutte le informazioni inviate dagli agent installati, con un approccio proattivo e predittivo alla gestione degli eventi sfruttando i dati di utilizzo dei sistemi e delle applicazioni, permette la correlazione dei dati di tutti gli anelli della catena tecnologica che concorrono all'erogazione del servizio, supportando la Root Cause Analysis

Nagios XI è lo strumento adottato per il monitoraggio della rete e dell'infrastruttura IT dell'Istituto. La soluzione è in grado di monitorare tutti i componenti critici dell'infrastruttura, tra cui database, sistemi operativi, protocolli di rete. Lo strumento permette inoltre di eseguire il monitoraggio della rete sia in termini di componenti infrastrutturali (es: switch) che in termini di qualità del traffico di rete e di banda dell'intera infrastruttura IT, in modo da verificare che sistemi, applicazioni e servizi funzionino correttamente.

Lo strumento è integrato anche con altre fonti esterne di monitoraggio quali SRM per il monitoraggio dello storage e Schneider per il monitoraggio degli eventi di natura ambientale dei DataCenter Inail.

Apache JMeter è lo strumento utilizzato per testare le prestazioni sia su risorse statiche che dinamiche delle applicazioni Web. Lo strumento viene utilizzato per predisporre il monitoraggio delle navigazioni simulate per consentire di individuare un malfunzionamento anche quando il carico sui servizi è minimo o nullo; questo permette un monitoraggio proattivo con la possibilità di evidenziare eventuali errori nei servizi prima degli utenti finali.