

Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA "TULLIO LEVI-CIVITA"

CORSO DI LAUREA IN INFORMATICA



Realizzazione a microservizi di una
applicazione di process mining per il
filtraggio degli eventi

Tesi di laurea triennale

Relatore

Prof. Tullio Vardanega

Laureando

Fontolan Carlo

Lorem ipsum dolor sit amet, consectetur adipiscing elit.

— Oscar Wilde

Dedicato a ...

Sommario

Il presente documento descrive il lavoro svolto durante il periodo di stage, dalla durata di circa trecentoquaranta ore, del laureando Carlo Fontolan presso l'azienda Siav S.p.A. Lo studente vuole descrivere con riflessioni critiche e oggettive quanto appreso e maturato dal punto di vista professionale e delle competenze in tutto l'arco di durata dello stage. In primo luogo era richiesto lo sviluppo di una libreria di filtraggio in abito process mining, che verrà poi inserita all'interno di una architettura a microservizi. In secondo luogo, è stata richiesta la riscrittura di alcune sezioni di un loro vecchio prodotto, in cui la libreria sopradescritta farà da protagonista, per quanto riguarda la sezione di filtraggio. Le caratteristiche richieste dall'azienda proponente erano che il prodotto fosse idoneo a girare su cloud e che fosse costituito da un'architettura a microservizi e sviluppato con framework all'avanguardia. Siccome in corso d'opera sono sorte alcune criticità in merito allo sviluppo della libreria, in accordo con il tutor aziendale, abbiamo deciso di rinegoziare alcuni requisiti andando ad eliminare la sezione riguardante il *backend*, facendo spazio ad alcuni stub in modo da poter osservare il comportamento dell'interfaccia, soddisfacendo così gli obiettivi sia a livello aziendale che personale.

Ringraziamenti

Innanzitutto, vorrei esprimere la mia gratitudine al Prof. Tullio Vardanega, relatore della mia tesi, per l'aiuto e il sostegno fornitomi durante la stesura del lavoro.

Desidero ringraziare con affetto i miei genitori per il sostegno, il grande aiuto e per essermi stati vicini in ogni momento durante gli anni di studio.

Ho desiderio di ringraziare poi i miei amici per tutti i bellissimi anni passati insieme e le mille avventure vissute.

Padova, Dicembre 2019

Fontolan Carlo

Indice

1	Il contesto aziendale	1
1.1	L'azienda	1
1.2	Tecnologie utilizzate	2
1.2.1	<i>Frontend</i>	2
1.2.2	<i>Backend</i>	2
1.3	Processi aziendali	3
1.3.1	Metodologie di sviluppo	3
1.3.2	Strumenti di supporto	3
1.4	Clientela rivolta a Siav	4
1.4.1	Propensione all'innovazione	4
2	L'attività di stage all'interno della strategia aziendale	5
2.1	Vantaggi aziendali	5
2.2	Introduzione al progetto TODO	6
2.2.1	Analisi software attuale	6
2.2.2	Progettazione libreria in relazione all'architettura finale	6
2.2.3	Progettazione interfaccia <i>frontend</i>	7
2.3	Aspettative Aziendali	8
2.3.1	Vincoli	9
2.4	Aspettative Personali	11
3	Resoconto dello stage	13
3.1	Introduzione al progetto	13
3.2	Pianificazione del progetto	13
3.3	Analisi dei requisiti	14
3.4	Progettazione	14
3.4.1	Libreria	14
3.4.2	Frontend	14
3.5	Codifica	14
3.6	Verifica	14
3.7	Validazione	14
3.8	Copertura	14
4	Valutazione Retrospettiva	15
4.1	Soddisfaccimento Obiettivi	15
	Glossario	17
	Acronimi	19

Elenco delle figure

1.1	I prodotti offerti da Siav	1
1.2	Diagramm illustrativo sul funzionamento di RabbitMQ	2
1.3	Ciclo di vita della metodologia Agile	3
2.1	Illustazione della differenza tra chiamate sincrone ed asincrone	7
2.2	Illustazione semplificata di un'architettura server cloud	8
2.3	Diagramma di gantt relativo alle attività di stage	9
2.4	Illustrazione semplificata di <i>Daily Meeting</i>	10

Elenco delle tabelle

2.1	Obiettivi obbligatori dello Stage	8
2.2	Obiettivi desiderabili dello Stage	8

Capitolo 1

Il contesto aziendale

1.1 L'azienda

Siav è una delle più importanti realtà italiane di sviluppo software e di servizi informatici specializzata nella dematerializzazione e nella gestione documentale e nei processi digitali. Presenta diverse filiali nel suolo italiano che cooperano tra loro per un fine comune. Siav inoltre punta molto sulla collaborazione tra aziende, in campo nazionale sia a livello pubblico che privato, ma anche a livello internazionale. Una fra tutti Microsoft. I loro servizi puntano in primo luogo, ad una miglior gestione, controllo ed automazione di tutti i principali processi aziendali; in secondo luogo, ad offrire diverse soluzioni concrete anche a singoli privati che necessitano di sistemi di gestione ed integrazione per la loro piccola realtà. Siav si colloca all'interno del mercato come una



Figura 1.1: I prodotti offerti da Siav

tra la più importanti realtà italiane a livello di *Enterprise Content Management* offrendo

servizi per poter migliorare processi e gestioni aziendali spaziando da un gestore di caselle PEC, ad un applicativo di process mining denominato *Bipod*, fino ad arrivare al loro prodotto di punta: Archiflow/Sillog: un software in continuo sviluppo, mantenuto e aggiornato da piccoli team che lavorano in sinergia per raggiungere un obiettivo comune. Archiflow/Sillog offre una soluzione alla gestione di una cospicua mole di documenti, categorizzandoli in varie sezioni permettendone un facile reperimento.

1.2 Tecnologie utilizzate

Le tecnologie utilizzate dell'azienda per la realizzazione dei propri prodotti sono fondate principalmente sulla possibilità di un utilizzo tramite cloud. Si sono dedicati principalmente sullo sviluppo di *Webapp*. Tali tecnologie possono essere raggruppate in due macrocategorie: *Frontend* e *Backend*.

1.2.1 Frontend

Da quel che ho potuto constatare durante l'attività di stage, per quanto riguarda lo sviluppo web lato *client* viene utilizzata prevalentemente la piattaforma Angular. Tramite il consistente numero di componenti prefabbricati presenti in rete e una vasta gamma di funzionalità e servizi disponibili, è possibile sviluppare un'interfaccia frontend di qualità a proprio piacimento, soddisfacendo nel miglior modo possibile le aspettative del cliente.

1.2.2 Backend

Per quanto riguarda lato backend vengono utilizzate molteplici tecnologie in base alle necessità che prevede il software di sviluppo; *RabbitMQ*: questa tecnologia è stata pensata principalmente per un'architettura a microservizi e viene utilizzata in più prodotti all'interno dell'azienda. *RabbitMQ* viene definito come un *Broker* di messaggistica: ossia un sistema che monitora la trasmissioni di messaggi tra servizi. Un'altro sistema utilizzato, restano in ambito di microservizi è *Kubernetes*: un sistema per la gestione di *container*.

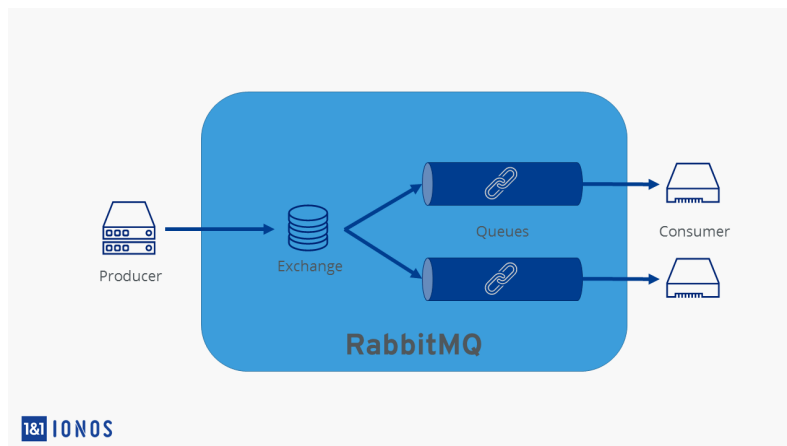


Figura 1.2: Diagramm illustrativo sul funzionamento di RabbitMQ

1.3 Processi aziendali

1.3.1 Metodologie di sviluppo

L'azienda normalmente si ritrova a dover far fronte ad alcune problematiche derivate da bug, criticità o variazioni importanti di requisiti che possono essere riscontrate durante i vari processi aziendali ed incidere in modo significativo sull'andamento dello sviluppo. Per cercare di venire incontro a ciò l'azienda ha adottato una metodologia di tipo *Agile*, cercando di mantenere un atteggiamento flessibile rispetto ai processi e gli obiettivi. Risulta quindi fondamentale l'organizzazione di incontri durante vari momenti della giornata o settimana in modo da confrontarsi sull'andamento dei processi e lo stato di avanzamento del prodotto. È consuetudine dei team di lavoro un confronto giornaliero tramite *daily meeting*, per discutere su quanto fatto durante la giornata precedente indicando le eventuali criticità riscontrate, pianificando così gli obiettivi per la giornata odierna. Ad inizio settimana invece viene organizzato *weekly meeting* per constatare gli stati di avanzamento rispetto alla settimana precedente, per poi fissare gli obiettivi per la settimana successiva. Gli obiettivi, le attività preposte e le scadenze prefissate venivano solitamente appesi in una lavagna posta in luogo ben visibile all'interno del reparto, per ordine di priorità; in questo modo era quindi possibile tener sott'occhio le principali attività. Il rapporto con il cliente è di vitale importanza per *Siav*. L'azienda infatti presenta un servizio di assistenza clienti tramite il quale vengono effettuate segnalazioni di *bug* o assistenza sui propri prodotti

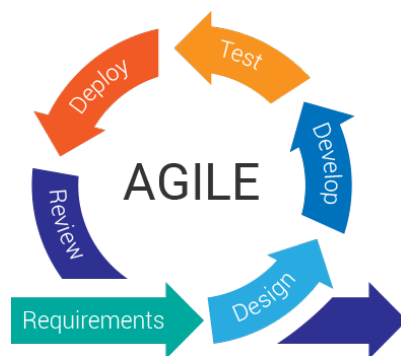


Figura 1.3: Ciclo di vita della metodologia Agile

1.3.2 Strumenti di supporto

L'azienda mette a disposizione diversi strumenti di supporto, allo scopo di gestire e tenere traccia nel migliore dei modi tutte le attività di progetto. Facendo fede alla metodologia *Agile*, la maggior parte degli strumenti servono a gestire tutti gli aspetti che riguardano il codice. Viene utilizzato *TFS*, uno strumento di *Microsoft* per la gestione delle principali attività di progetto che offre un set di strumenti per la collaborazione. Tra le principali funzionalità sono presenti: gestione del repository tramite tecnologie Git o *TFVC*, gestione dei requisiti e strumenti di *build* automatizzati. Tale strumento aderisce in maniera completa a alle tecniche *agile* andando quindi ad integrarsi in modo solido all'interno della realtà lavorativa. Per quanto riguarda il tracciamento delle attività viene utilizzato *Evernote*: un software che permette la scrittura e la condivisione di note all'interno di un gruppo di utenti. Così facendo ogni

membro del team di sviluppo ha sotto controllo ogni attività svolta dagli altri membri. Un ulteriore strumento utilizzato per il tracciamento delle attività è *Google Docs*. Solitamente i task assegnati presentano una struttura semplice: la data di creazione, la possibilità o meno di menzionare direttamente l'interessato a cui risulta assegnato il task ed un casella di risposta per poter descrivere la sua effettiva terminazione, oppure un messaggio di testo di altro genere, che solitamente indica le problematiche per cui il task non è stato possibile risolverlo.

1.4 Clientela rivolta a Siav

La maggior parte dei clienti che si affidano a Siav sono aziende che presentano il bisogno di automatizzarsi e migliorare i propri processi interni andando così ad incrementare la propria efficienza sotto l'aspetto lavorativo. Tali aziende sono di vario genere, dal semplice ristorante ad una nota catena di supermercati fino ad aziende metalmeccaniche. Per ogni settore, Siav offre diverse opportunità di gestione e miglioramento aziendale. Siav è catalogata come *Software house* e presenta un ampio catalogo di prodotti atti a soddisfare le principali necessità organizzative di un'azienda, sta al potenziale cliente poi, valutarne l'acquisto in base alle proprie necessità.

1.4.1 Propensione all'innovazione

Siav è un'azienda che fa dell'innovazione un suo punto cardine. I prodotti che offre sono costantemente aggiornati, cercando di adattarsi alle necessità del cliente. Per poter far ciò l'azienda ha messo a disposizione un servizio di *Helpdesk* in modo da fornire assistenza in merito ai propri prodotti per i propri clienti. L'azienda inoltre è alla continua ricerca di nuove tecnologie da poter integrare all'interno dei propri applicativi; è presente all'interno dell'organico un team di ricerca e sviluppo che analizza, le varie piattaforme e tecnologie presenti sul mercato. Spesso e volentieri tali attività vengono contestualizzate all'interno di proposte di stage, in modo da poter osservare nel concreto se le ricerche fatte in precedenza possano portare a benefici per i loro prodotti, facendo anche un'attenta analisi dei costi.

Capitolo 2

L'attività di stage all'interno della strategia aziendale

Introduzione al capitolo

In questo capitolo si andrà ad illustrare la propensione dell'azienda rispetto allo stage, indicando le attività con le relative tematiche che mi sono state proposte e come tali siano inserite all'interno della strategia aziendale

2.1 Vantaggi aziendali

Siav da molti anni ha stretto un accordo con l'Università di Padova per quanto riguarda la propensione a svolgere attività di stage. È in continua ricerca di nuove figure da poter inserire all'interno del loro organigramma, soprattutto per quanto riguarda studenti o neo laureati. Essendo un'azienda in forte crescita ed espansione, necessita costantemente di nuovo personale, ma soprattutto, di nuove idee e tecnologie da poter poi mettere in pratica nei loro prodotti. La maggior parte degli stage viene attuata dal settore di Ricerca e Sviluppo proprio per soddisfare questa necessità; solitamente le attività di stage proposte comprendono un periodo di formazione iniziale per poter prendere dimestichezza con le nuove tecnologie che si andranno ad attuare, per poi metterle in pratica con il supporto costante del team. Questa sezione presente all'interno di *Siav* garantisce una costante dedizione alla ricerca, non soffermandosi a ciò che è già presente all'interno dell'azienda, ma cercando di spaziare tra più tematiche in contemporanea, procedendo all'attivazione di più stage in ambiti diversi, spaziando quindi la ricerca su più fronti. L'azienda, inoltre, avendo stipulato un cospicuo numero di *partnership* si ritrova spesso e volentieri a partecipare a diversi *meeting* o seminari per quanto riguarda lo studio e approfondimento di nuove tecnologie emergenti o di tendenza. Tali aspetti vengono poi valutati dal team tramite un studio di fattibilità per poter verificare se le tematiche affrontate possono portare qualcosa di redditizio all'interno del contesto aziendale. Quest'ultime portano *Siav* ad essere una delle realtà italiane più attive e propense alla ricerca e allo sviluppo.

2.2 Introduzione al progetto TODO

Siav attraverso i numerosi progetti di stage mira ad esplorare nuove tematiche e nuove opportunità tecnologiche che possano portare ad un miglioramento dei propri prodotti. Il progetto che mi è stato proposto mirava proprio al miglioramento di un loro applicativo. Lo scopo dello stage, è stato diviso in tre parti:

- * sviluppo di una libreria di *process mining* per la pulizia di log degli eventi
- * sviluppo di un'interfaccia *frontend* che rispettasse le tematiche presenti all'interno della libreria
- * sviluppo di stub che simulino il comportamento del *backend* per verificare il corretto funzionamento dell'interfaccia.

Tramite questo stage è stato possibile per *Siav* migliorare uno dei propri prodotti che aveva perso la sua mantenibilità, essendo stato sviluppato da un piccolo team che ora non era più presente in azienda. Oltretutto la documentazione è risultata scarsa e di difficile comprensione; è stato quindi deciso di riscrivere l'applicativo da zero, cercando di mantenere le principali funzionalità, analizzando qualche aspetto critico che presentava la vecchia architettura e cercando un'alternativa solida basandosi su nuove tecniche e tecnologie che all'epoca non erano state prese in considerazione.

2.2.1 Analisi software attuale

Durante la prima parte dello stage l'obiettivo è quello di analizzare le funzionalità e interfaccia legate al filtraggio dei log presente nell'applicativo in modo da poter individuare tutte le possibili funzionalità disponibili all'utente. Dopo una revisione completa ed approfondita dell'applicativo ne è emerso che dal punto di vista delle funzionalità l'applicativo non aveva nulla da invidiare rispetto ai più comuni software di *process mining*, presentando tutte le principali funzionalità per la modellazione di log. Dal punto di vista della sua efficienza però, le cose andavano diversamente: tutto il sistema era gestito tramite chiamate sicrone; ciò comportava la presenza di lunghi tempi di attesa per ogni operazione che l'utente intendeva effettuare. Solamente per effettuare un'operazione di filtraggio, per un log della dimensione di qualche *Megabyte*, era necessario un tempo di attesa che andava da 1 a 5 secondi. Il che può essere accettabile, se non fosse che in ambito *process mining*, per poter gestire di un cospicuo numero di processi, vengono analizzati log che possono essere di gran lunga più corposi e impegnativi da analizzare per il sistema. Da ciò ne è scaturita l'idea di una restaurazione dell'applicativo.

2.2.2 Progettazione libreria in relazione all'architettura finale

In seguito all'attività di analisi del vecchio applicativo un obiettivo dello stage risiede nella progettazione di una libreria di filtraggio in grado di poter effettuare le principali operazioni di pulizia su log degli eventi. Tale libreria dovrà soddisfare alcuni requisiti fondamentali:

- * dovrà essere mantenibile nel tempo, facendo uso quindi di opportuni principi cardine della Programmazione ad Oggetti e dovrà essere documentata in modo preciso in modo che chiunque, in futuro, la analizzi per poter effettuare alcune modifiche possa apprendere in modo chiaro e trasparente ogni sua parte e la sua struttura.

- * dovrà essere affidabile e testata in ogni sua parte tramite *Unit Test* in modo da garantire una base solida di partenza per poi contestualizzarla all'interno di un'architettura a microservizi

2.2.3 Progettazione interfaccia *frontend*

Successivamente all'implementazione della libreria sarà necessario progettare l'interfaccia *fronted*; per quest'ultima sarà necessario avvalersi del loro vecchio applicativo *Bipod*, per poi confrontarlo con la libreria sviluppata discutendo con il tutor eventuali modifiche in caso di discrepanze. Tale interfaccia sarà rivista in modo più preciso ed approfondito una volta terminata la libreria, avendo quindi una visuale più ampie delle funzionalità offerte. Inoltre dovrà essere sviluppata con interfaccia più intuitiva e semplificata rispetto a quella presente sull'applicativo odierno.

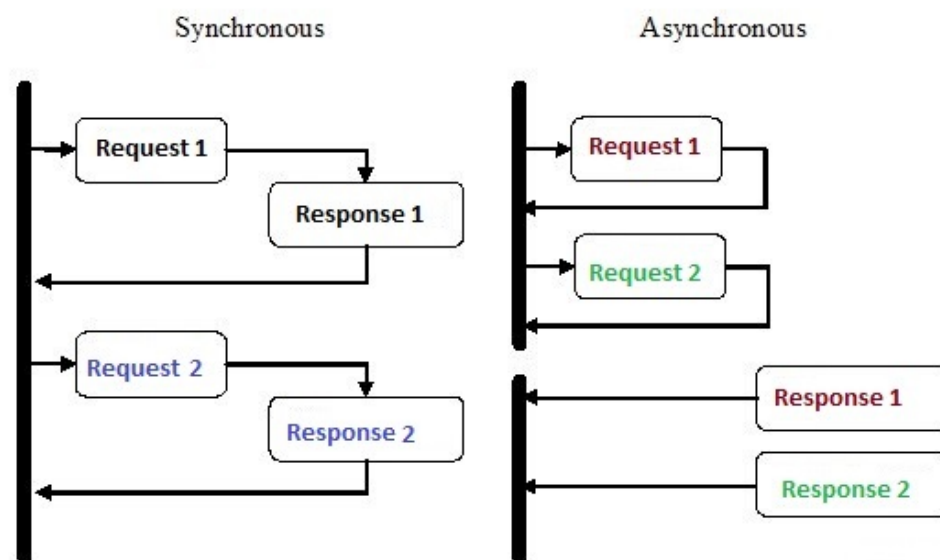


Figura 2.1: Illustrazione della differenza tra chiamate sincrone ed asincrone

2.3 Aspettative Aziendali

Al termine delle 340 ore l'azienda si aspetta di avere un libreria di filtraggio solida ed estendibile, in grado di eseguire le principali funzionalità di filtraggio all'interno di un log degli eventi e un'interfaccia *frontend* semplice ed intuitiva in grado di utilizzare tutte le funzionalità incluse all'interno della libreria. Il tutto deve essere pensato per un'erogazione in modalità *cloud*, pertanto le scelte architetturali dovranno tener conto di questo prerequisito. Tali attività mi hanno portato a definire i principali requisiti



Figura 2.2: Illustrazione semplificata di un'architettura server cloud

con il tutor aziendale, che sono poi stati inseriti all'interno del piano di lavoro. In seguito a ciò il tutor ha individuato gli obiettivi per il progetto suddividendoli in due categorie: obbligatori e desiderabili.

Tabella 2.1: Obiettivi obbligatori dello Stage

Obiettivi obbligatori
Inquadramento del problema: stesura di un documento di specifica del problema affrontato;
Analisi dei requisiti e specifiche tecniche di progettazione: stesura dei relativi documenti;
Implementazione del frontend web e dei servizi di backend necessari a gestire le funzionalità di filtraggio degli eventi
Collaudo del sistema: il progetto deve prevedere una fase di test del software implementato, con documentazione dei risultati ottenuti

Tabella 2.2: Obiettivi desiderabili dello Stage

Obiettivi desiderabili
point-and-click sul processo per inserimento di filtri sulle successioni di eventi
implementazione test frontend

2.3.1 Vincoli

I vincoli che mi sono stati posti durante la stesura del piano di lavoro possono essere raggruppati in 3 categorie:

Vincoli temporali

Rappresentano i vincoli dal punto di vista del tempo. Lo svolgimento dello stage ha avuto una durata di 340 ore. Tali ore sono state distribuite in modo uniforme durante l'arco di 9 settimane. Inizialmente sono state concordate 8 settimane, ma in seguito ad alcune criticità riscontrate in fase di sviluppo, con la conseguente negoziazione dei requisiti, sono state portate a 9; includendo quindi una settimana aggiuntiva per terminare tutte le attività concordate. Nel periodo antecedente all'inizio dello stage l'azienda ha redatto una scansione temporale delle attività su base settimanale suddivisa nel seguente modo:

- * **Settimana 1:** Studio introduttivo del Process Mining con le relative tecnologie. Studio dell'architettura di storage messa a disposizione e del framework ProM
- * **Settimana 2:** Studio di fattibilità per la realizzazione dei moduli necessari a erogare le funzionalità di filtraggio
- * **Settimana 3-4-5:** Progettazione dei moduli di discovery del processo, visualizzazione log e filtraggio
- * **Settimana 6-7:** Implementazione del software e deploy su docker
- * **Settimana 8:** Test e sperimentazione del software

Durante tutto l'arco delle settimane è stata richiesta anche una documentazione sul lavoro includendo una breve cronologia delle attività svolte, indicando le eventuali criticità riscontrate e specificando in che modo sono state affrontate e risolte. Tale pianificazione è stata rinegoziata a partire dalla sesta settimana; in quanto l'architettura finale non era stata ancora ideata in modo specifico. Questo ha portato, in accordo con il tutor ad una rinegoziazione dei requisiti e quindi della attività programmate; sostituendo il *deploy* su *docker* con l'implementazione di alcuni stub in modo da poter effettuare test lato frontend.

#	Attività	Settimane							
		1	2	3	4	5	6	7	8
1	Studio introduttivo PM								
2	Studio librerie								
3	Studio di fattibilità								
4	Analisi dei requisiti								
5	Progettazione web service e GUI								
7	Implementazione del <i>software</i>								
8	Test e sperimentazione del <i>software</i>								
9	Documentazione								

Figura 2.3: Diagramma di gantt relativo alle attività di stage

Vincoli metodologici

Lo stage è stato svolto presso la sede di Rubano(PD) in comune accordo con il tutor. Ciò è stato deciso con lo scopo di poter confrontarsi ed interagire in maniera sistematica ogni qualvolta ce ne fosse stato bisogno; non soltanto tramite un rapporto stagista e tutor, ma confrontandosi con altre figure presenti all'interno dell'azienda in modo da poter consolidare nuovi rapporti di tipo professionale, in modo da avere supporto aggiuntivo in caso ce ne fosse stata necessità. Oltre a ciò è stato richiesto un continuo confronto sulle attività effettuate e quelle in programma tramite alcune note di testo condivise tra il tutor ed il team di Ricerca e Sviluppo. Tali note sono poi oggetto di discussione tramite *Daily meeting*: in cui vi è un momento di confronto tra i membri dei team per discutere in merito alle attività svolte e quelle in programma per la giornata. Mi stata messa a disposizione una postazione di lavoro con personal computer, connessione ad Internet ed alla rete locale con accesso al server di sviluppo, in modo da poter effettuare *deploy* di eventuali macchine virtuali necessarie allo sviluppo. Sono stati forniti una serie dati di esempio per testare quanto sviluppato, oltre a tutte le librerie necessarie allo sviluppo stesso dell'applicativo.

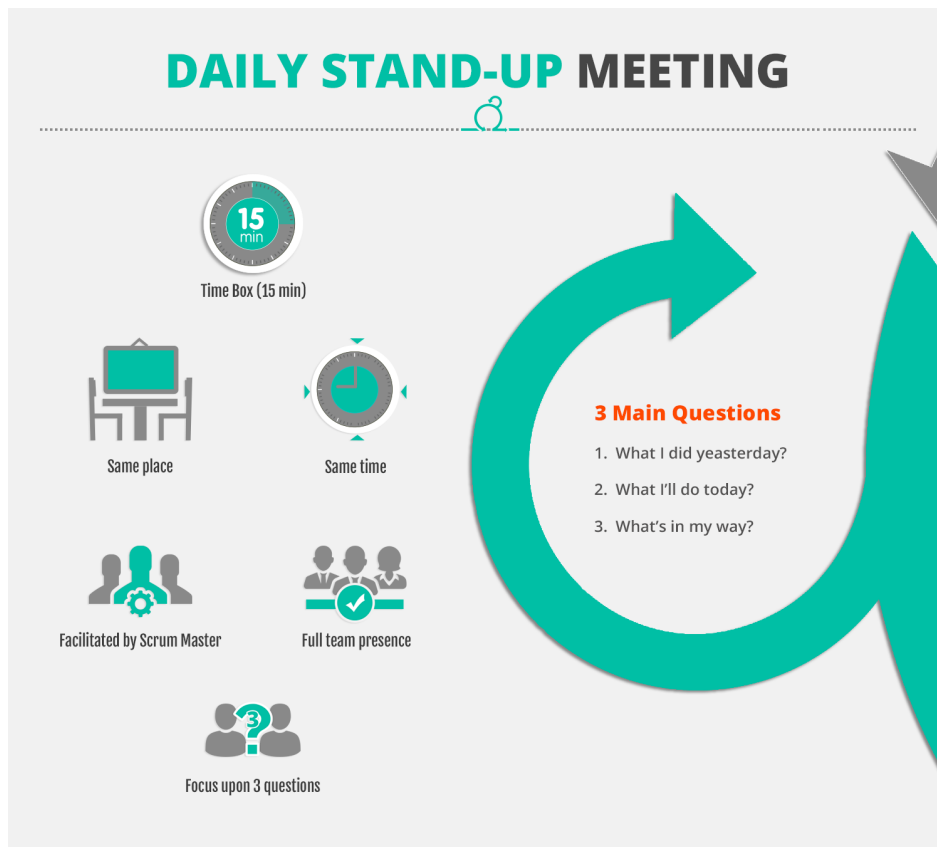


Figura 2.4: Illustrazione semplificata di *Daily Meeting*

Vincoli tecnologici

Come sopra descritto l'obiettivo fondamentale per l'azienda è che l'applicativo finale fosse in grado di girare su server *Cloud*. Questo ha influito in modo significativo sui vincoli tecnologici che sono stati imposti. Innanzitutto tramite lo sviluppo di una libreria che fosse scritta in modo semplice e leggibile, per far sì che chi dovrà inserirla nell'ambiente a microservizi abbia un'idea precisa delle sue caratteristiche ed il suo funzionamento; in secondo luogo tramite la gestione dell'asincronia all'interno dell'interfaccia *frontend*, andando a gestire i messaggi in arrivo del server, categorizzandoli e cambiando il comportamento dell'interfaccia a seconda della caratterizzazione del messaggio.

2.4 Aspettative Personali

Tutte le mie passate esperienze sono riconducibili esclusivamente all'ambito accademico; l'esperienza che più mi ha avvicinato all'ambito lavorativo è stato il progetto di Ingegneria del Software; tramite il quale ho potuto constatare alcune dinamiche presenti all'interno di un contesto professionale. Per poter avvicinarmi al mondo lavorativo ho partecipato all'evento organizzato dall'Università di Padova *Stage-IT* tramite il quale ho potuto affrontare diversi incontri con realtà aziendali ben formate e strutturate. Durante la fiera ho potuto valutare diverse attività di stage tenendo in considerazione le tematiche trattate e quanto ciò potessero incidere nel mio grado di formazione. La maggior parte di esse non sono risultate, a mio parere, stimolanti per la mia maturità a livello professionale. Le aspettative che mi sono preposte prima dell'evento di *Stage-IT* sono state le seguenti:

- * Apprendimento di nuove metodologie di lavoro
- * Visione e apprendimento di nuove tecnologie
- * Lavoro in modo autonomo sotto la supervisione che tutor che mi presti supporto
- * Visione del contesto lavorativo in maniera più ampia, non legata strettamente all'attività di stage

Successivamente a *Stage-IT* sono stato contattato da altre aziende operanti nel settore dell'informatica, ognuna delle quali mi ha proposto varie attività affrontando diversi argomenti e tecnologie. Una su tutte che mi ha colpito è stata proprio *Siav* che, attraverso le loro proposte e la loro costante dedizione in merito alla ricerca di nuove tematiche e tecnologie mi ha convinto a svolgere l'attività di stage presso la loro azienda, collocandomi proprio all'interno del team di Ricerca e Sviluppo. Ciò ha comportato un aumento delle mie aspettative di fronte a questa attività:

- * Valutare quanto possano essermi state utili le nozioni apprese durante il percorso universitario.
- * apprendimento di nuove tecnologie
- * Apprendimento dei principi cardine del process mining
- * Apprendimento sul funzionamento di un'architettura a microservizi
- * Instaurazione di discussioni con il personale in modo da potersi confrontare ed avere diversi punti di vista rispetto ad alcune tematiche

Capitolo 3

Resoconto dello stage

In questo capitolo andrò a descrivere i principali aspetti che sono stati affrontati durante lo stage

3.1 Introduzione al progetto

Visto in scala più ampia, il progetto finale risulterà una rivisitazione di Bipod, un applicativo di Siav, in ambito process mining, utilizzato come gestore di processi aziendali. Tale software, dopo averlo provato con mano, risulta funzionale in ogni sua parte, anche se la sua struttura risulta poco estendibile. Per questo motivo mi è stato proposto di sviluppare una parte di software che si andrà poi ad integrare con il nuovo applicativo che andrà a rimpiazzare l'attuale esistente. Nello specifico le richieste a cui ho fatto fronte sono state le seguenti:

- * Libreria di process mining per filtraggio su log degli eventi.
- * Interfaccia fronted tenendo come punto di riferimento il vecchio applicativo.
- * Stub per verificare il corretto comportamento dell'interfaccia di filtraggio

3.2 Pianificazione del progetto

Durante il periodo antecedente all'inizio dell'attività di progetto sono state discusse con il tutor tutte le principali attività che avrei dovuto svolgere nell'arco dei due mesi preposti. Tali attività, anche se in forma generica, sono state inserite nel diagramma di Gantt presente al capitolo precedente (Figura 2.3). Per tener traccia di tutte le attività svolte durante il periodo di stage, tramite il supporto del tutor e del team in cui sono stato inserito ho cercato di apprendere nella miglior maniera possibile i principi cardine dell'*Agile programming*. Non avendo mai avuto l'occasione prima d'ora di avvicinarmi a tale metodologia è stato impegnativo, ma appagante entrare nei suoi meccanismi pratici e nelle sue dinamiche. Tuttavia sono riuscito, almeno in parte, a comprendere alcune importanti metodologie che sono state applicate durante tutto il periodo di stage. Le attività sono state tracciate tramite semplici note, condivise con l'interno team, in cui è descritta una breve cronologia di tutte le attività svolte giorno per giorno, indicando eventuali problematiche riscontrate e come sono state affrontate.

3.3 Analisi dei requisiti

3.4 Progettazione

3.4.1 Libreria

3.4.2 Frontend

3.5 Codifica

3.6 Verifica

3.7 Validazione

3.8 Copertura

Capitolo 4

Valutazione Retrospettiva

Breve introduzione al capitolo

4.1 Soddisfacimento Obiettivi

Glossario

TFS Team foundation server. 3

TFVC Team foundation version control. 3

UML in ingegneria del software *UML*, *Unified Modeling Language* (ing. linguaggio di modellazione unificato) è un linguaggio di modellazione e specifica basato sul paradigma object-oriented. L'*UML* svolge un'importantissima funzione di “lingua franca” nella comunità della progettazione e programmazione a oggetti. Gran parte della letteratura di settore usa tale linguaggio per descrivere soluzioni analitiche e progettuali in modo sintetico e comprensibile a un vasto pubblico. 19

Acronimi

UML [Unified Modeling Language](#). 13, 25

Bibliografia