

Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA "TULLIO LEVI-CIVITA"

CORSO DI LAUREA IN INFORMATICA



## Realizzazione di una applicazione di process mining per il filtraggio degli eventi

*Tesi di laurea triennale*

*Relatore*

Prof. Tullio Vardanega

*Laureando*

Fontolan Carlo

---

ANNO ACCADEMICO 2018-2019



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. NomeDelProfessore, 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

<b>1</b>	<b>Introduzione</b>	<b>1</b>
1.1	L'azienda . . . . .	1
1.2	L'idea . . . . .	1
1.3	Organizzazione del testo . . . . .	2
<b>2</b>	<b>Processi e metodologie</b>	<b>3</b>
2.1	Processo sviluppo prodotto . . . . .	3
2.1.1	Metodologia Agile . . . . .	3
2.1.2	Metodologia Agile in relazione allo stage . . . . .	4
<b>3</b>	<b>Descrizione dello stage</b>	<b>5</b>
3.1	Introduzione al progetto . . . . .	5
3.2	Analisi preventiva dei rischi . . . . .	5
3.3	Requisiti e obiettivi . . . . .	5
3.4	Pianificazione . . . . .	5
<b>4</b>	<b>Analisi dei requisiti</b>	<b>7</b>
4.1	Casi d'uso . . . . .	7
4.2	Tracciamento dei requisiti . . . . .	8
<b>5</b>	<b>Progettazione e codifica</b>	<b>11</b>
5.1	Tecnologie e strumenti . . . . .	11
5.2	Ciclo di vita del software . . . . .	11
5.3	Progettazione . . . . .	11
5.4	Design Pattern utilizzati . . . . .	11
5.5	Codifica . . . . .	11
<b>6</b>	<b>Verifica e validazione</b>	<b>13</b>
<b>7</b>	<b>Conclusioni</b>	<b>15</b>
7.1	Consuntivo finale . . . . .	15
7.2	Raggiungimento degli obiettivi . . . . .	15
7.3	Conoscenze acquisite . . . . .	15
7.4	Valutazione personale . . . . .	15
<b>A</b>	<b>Appendice A</b>	<b>17</b>
	<b>Bibliografia</b>	<b>21</b>

# Elenco delle figure

1.1	Logo di Siav Spa . . . . .	1
2.1	Ciclo dei vita della metodolia agile . . . . .	3
4.1	Use Case - UC0: Scenario principale . . . . .	7

# Elenco delle tabelle

4.1	Tabella del tracciamento dei requisiti funzionali . . . . .	9
4.2	Tabella del tracciamento dei requisiti qualitativi . . . . .	9
4.3	Tabella del tracciamento dei requisiti di vincolo . . . . .	9

# Capitolo 1

## Introduzione

### 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 e cooperano tra loro per un fine comune. All'interno dello sede di sviluppo di Rubano erano presenti diversi reparti, ognuno con una mansione ed un scopo ben preciso dedicati ai vari software che fornisce l'azienda ai propri clienti. Essendo stato inserito sin da subito all'interno del team di ricerca e sviluppo (di cui il mio tutor ne è team leader) ho potuto vedere nel concreto la costante crescita ed espansione dell'azienda, che va ricercando costantemente nuove idee e strategie che spesso vengono proposte tramite attività di stage.



**Figura 1.1:** Logo di Siav Spa

### 1.2 L'idea

Il process mining è una disciplina di Business Intelligence per l'analisi ed il monitoraggio di processi aziendali. In particolare, le tecniche di process mining utilizzano analisi statistiche per estrarre informazioni utili dai dati di log di un'azienda. La pulizia dei dati in ingresso è un'operazione fondamentale di qualsiasi workflow di analisi. Questo è valido anche nel caso del Process Mining, dove ne abbiamo bisogno per due motivi principali: eliminare eventuali dati superflui dagli eventi raccolti e restringere la nostra analisi a sottocasi ben precisi, consentendo di portare il focus sulle prospettive desiderate. Da tempo l'azienda aveva sviluppato un applicativo in grado di effettuare molteplici operazioni su log degli eventi, anche se mal scritto e poco estendibile. L'idea che mi è stata proposta è stata quella di realizzare una libreria, che sarà successivamente inserita in un'architettura a microservizi, per poter effettuare operazioni di filtraggio, puntando soprattutto sull'estendibilità e mantenibilità del codice in modo da poter rimpiazzare in futuro l'applicativo esistente. L'idea in sé è stata molto apprezzata sia per quanto riguarda l'argomento del process mining, sia per le nuove tecnologie che ho utilizzato.

### 1.3 Organizzazione del testo

**Il secondo capitolo** descrive le tecniche e le metodologie adottate per lo sviluppo del progetto

**Il terzo capitolo** approfondisce ...

**Il quarto capitolo** approfondisce ...

**Il quinto capitolo** approfondisce ...

**Il sesto capitolo** approfondisce ...

**Nel settimo capitolo** descrive ...

Riguardo la stesura del testo, relativamente al documento sono state adottate le seguenti convenzioni tipografiche:

- \* gli acronimi, le abbreviazioni e i termini ambigui o di uso non comune menzionati vengono definiti nel glossario, situato alla fine del presente documento;
- \* per la prima occorrenza dei termini riportati nel glossario viene utilizzata la seguente nomenclatura: *parola*<sup>[g]</sup>;
- \* i termini in lingua straniera o facenti parti del gergo tecnico sono evidenziati con il carattere *corsivo*.

## Capitolo 2

# Processi e metodologie

### *Introduzione al capitolo*

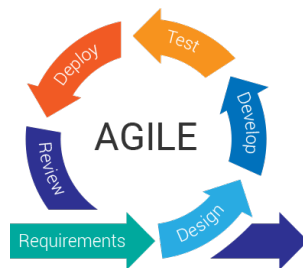
In questo capitolo andranno affrontati i processi e le metodologie affrontate per lo sviluppo del progetto. Più precisamente verranno descritti i principi cardine dell'*Agile programming* e successivamente verrà illustrato come questi siano stati applicati durante l'attività di stage.

## 2.1 Processo sviluppo prodotto

Lo stage presso Siav mi ha permesso di entrare concretamente a contatto con una metodologia di gestione di progetto vista durante il corso di Ingegneria del Software: la metodologia Agile.

### 2.1.1 Metodologia Agile

La metodologia Agile si riferisce ad un insieme di metodi di sviluppo del software emersi nei primi anni 2000 e fondati su un insieme di principi comuni derivati dai principi del "Manifesto per lo sviluppo agile del software". Fra le pratiche presenti vi è la formazione di team di sviluppo piccoli, lo sviluppo iterativo e incrementale, la pianificazione adattiva e il rapporto costante e diretto con il proponente durante il processo di sviluppo.



**Figura 2.1:** Ciclo dei vita della metodolia agile

### 2.1.2 Metodologia Agile in relazione allo stage

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 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:

- \* Stand-up daily meeting : Ogni giorno, solitamente la mattina, vi era un momento di confronto con tutti i membri del team in merito alle attività da svolgere durante la giornata, tenendo in considerazione eventuali criticità che sarebbero potute emergere ed eventuali problematiche che sono state riscontrate nella giornata precedente.
- \* Stand-up weekly meeting: Ad inizio settimana veniva discussa con il tutor la pianificazione del lavoro per la settimana, analizzando se il percorso intrapreso fosse congruo con gli obiettivi preposti.
- \* Software over documentation: Durante tutto il periodo di progetto è stata data maggior priorità al software sviluppato piuttosto che alla documentazione. È stata racchiusa ogni macro attività all'interno di brevi note per poter tener traccia delle scelte fatte e delle attività svolte, prive di formalismi rispetto alla documentazione classica. Tramite ciò è stato possibile dare una maggior attenzione allo sviluppo, concentrandosi maggiormente sulle richieste del proponente, mantenendo però una cronologia delle attività svolte e dalla scelte adottate in modo sintetico ma preciso.
- \* Dialogo continuo con il cliente: Il cliente nel mio caso veniva individuato dal tutor aziendale (team leader del reparto R&D). Il dialogo è stato frequente e mirava ad aggiornarlo frequentemente riguardo allo stato di avanzamento dei processi per verificare se le attività che mi ero prefissato erano in linea con le aspettative.

## Capitolo 3

# Descrizione dello stage

*Breve introduzione al capitolo*

### 3.1 Introduzione al progetto

### 3.2 Analisi preventiva dei rischi

Durante la fase di analisi iniziale sono stati individuati alcuni possibili rischi a cui si potrà andare incontro. Si è quindi proceduto a elaborare delle possibili soluzioni per far fronte a tali rischi.

#### **1. Performance del simulatore hardware**

**Descrizione:** le performance del simulatore hardware e la comunicazione con questo potrebbero risultare lenti o non abbastanza buoni da causare il fallimento dei test.

**Soluzione:** coinvolgimento del responsabile a capo del progetto relativo il simulatore hardware.

### 3.3 Requisiti e obiettivi

### 3.4 Pianificazione





## Capitolo 4

# Analisi dei requisiti

*Breve introduzione al capitolo*

### 4.1 Casi d'uso

Per lo studio dei casi di utilizzo del prodotto sono stati creati dei diagrammi. I diagrammi dei casi d'uso (in inglese *Use Case Diagram*) sono diagrammi di tipo [Unified Modeling Language \(UML\)](#) dedicati alla descrizione delle funzioni o servizi offerti da un sistema, così come sono percepiti e utilizzati dagli attori che interagiscono col sistema stesso. Essendo il progetto finalizzato alla creazione di un tool per l'automazione di un processo, le interazioni da parte dell'utilizzatore devono essere ovviamente ridotte allo stretto necessario. Per questo motivo i diagrammi d'uso risultano semplici e in numero ridotto.



**Figura 4.1:** Use Case - UC0: Scenario principale

#### UC0: Scenario principale

**Attori Principali:** Sviluppatore applicativi.

**Precondizioni:** Lo sviluppatore è entrato nel plug-in di simulazione all'interno dell'I-DE.

**Descrizione:** La finestra di simulazione mette a disposizione i comandi per configurare, registrare o eseguire un test.

**Postcondizioni:** Il sistema è pronto per permettere una nuova interazione.

## 4.2 Tracciamento dei requisiti

Da un'attenta analisi dei requisiti e degli use case effettuata sul progetto è stata stilata la tabella che traccia i requisiti in rapporto agli use case.

Sono stati individuati diversi tipi di requisiti e si è quindi fatto utilizzo di un codice identificativo per distinguerli.

Il codice dei requisiti è così strutturato  $R(F/Q/V)(N/D/O)$  dove:

R = requisito

F = funzionale

Q = qualitativo

V = di vincolo

N = obbligatorio (necessario)

D = desiderabile

Z = opzionale

Nelle tabelle 4.1, 4.2 e 4.3 sono riassunti i requisiti e il loro tracciamento con gli use case delineati in fase di analisi.

**Tabella 4.1:** Tabella del tracciamento dei requisiti funzionali

Requisito	Descrizione	Use Case
RFN-1	L'interfaccia permette di configurare il tipo di sonde del test	UC1

**Tabella 4.2:** Tabella del tracciamento dei requisiti qualitativi

Requisito	Descrizione	Use Case
RQD-1	Le prestazioni del simulatore hardware deve garantire la giusta esecuzione dei test e non la generazione di falsi negativi	-

**Tabella 4.3:** Tabella del tracciamento dei requisiti di vincolo

Requisito	Descrizione	Use Case
RVO-1	La libreria per l'esecuzione dei test automatici deve essere riutilizzabile	-



# Capitolo 5

## Progettazione e codifica

*Breve introduzione al capitolo*

### 5.1 Tecnologie e strumenti

Di seguito viene data una panoramica delle tecnologie e strumenti utilizzati.

#### **Tecnologia 1**

Descrizione Tecnologia 1.

#### **Tecnologia 2**

Descrizione Tecnologia 2

### 5.2 Ciclo di vita del software

### 5.3 Progettazione

#### **Namespace 1**

Descrizione namespace 1.

**Classe 1:** Descrizione classe 1

**Classe 2:** Descrizione classe 2

### 5.4 Design Pattern utilizzati

### 5.5 Codifica



## Capitolo 6

# Verifica e validazione





## Capitolo 7

# Conclusioni

7.1 Consuntivo finale

7.2 Raggiungimento degli obiettivi

7.3 Conoscenze acquisite

7.4 Valutazione personale



Appendice A

Appendice A

Citazione

---

Autore della citazione







# Bibliografia