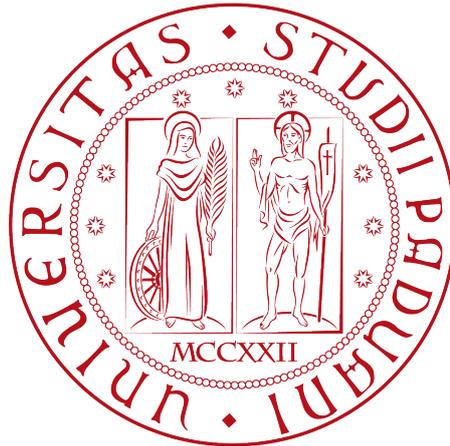


Università degli Studi di Padova
DIPARTIMENTO DI MATEMATICA "TULLIO
LEVI-CIVITA"
CORSO DI LAUREA IN INFORMATICA



Sviluppo e gestione di interfacce macchina in ambito certificazione Industria 4.0

Tesi di laurea Triennale

Relatore

Prof. Luigi De Giovanni

Laureando

Damiano D'Amico 1201216

ANNO ACCADEMICO 2022-2023

Sommario

L'Industria 4.0 è un'evoluzione della produzione industriale che utilizza tecnologie digitali al fine di creare una fabbrica intelligente. Per sfruttare appieno la metodologia di produzione inerente all'Industria 4.0, le fabbriche devono integrare i loro sistemi in una piattaforma unificata. Questa permette di raccogliere, analizzare ed utilizzare i dati per migliorare i processi di produzione. L'Industria 4.0 rappresenta una grande opportunità, le aziende che saranno capaci di sfruttarla potranno distinguersi dalla concorrenza continuando a crescere. Il presente documento descrive il lavoro svolto in questo ambito durante il periodo di stage, della durata di circa trecento ore, dal laureando Damiano D'Amico presso l'azienda Sanmarco Informatica S.p.A.

Convenzioni tipografiche

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

- * Gli acronimi ed i termini di uso non comune vengono definiti nel glossario, situato alla fine del documento. Essi saranno indicati solamente nella loro prima apparizione tramite il colore blu e l'apice "[g]";
- * I termini facenti parte del gergo tecnico sono evidenziati con il carattere in corsivo;
- * Viene fatto riferimento alla bibliografia tramite parentesi quadre contenenti il numero della citazione presente nella bibliografia.

Indice

1	Introduzione	1
1.1	Sanmarco Informatica S.p.A	1
1.1.1	Business Unit aziendale	2
1.1.2	Il team JMes	2
1.1.3	Framework SCRUM	3
1.2	L'Industria 4.0	5
1.2.1	Certificazione Industria 4.0	6
1.2.2	Manufacturing Execution System	7
1.2.3	Programmable Logic Controller	9
1.2.4	Gestione della memoria	9
1.3	Proposta di stage	10
1.3.1	Piano di lavoro	10
1.3.2	Gli obiettivi	10
2	Tecnologie utilizzate	13
2.1	Java	13
2.2	Tomcat	14
2.3	Synergy	15
2.3.1	Gestione delle personalizzazioni	17
2.4	GitLab	18
2.5	Eclipse	19
2.6	HeidiSQL	20
2.7	Overleaf	21

3	Software proprietari Sanmarco Informatica	23
3.1	Software JMes	23
3.2	Software JDI	24
3.3	Interfaccia JMes - JDI	25
4	Analisi e sviluppo: Interfacce e Personalizzazioni	27
4.1	Solmec	27
4.1.1	Le esigenze di Solmec	28
4.1.2	Sviluppo	28
4.2	Mimest	30
4.2.1	Le esigenze di Mimest	30
4.2.2	Sviluppo	30
4.2.3	Attività di supporto	31
4.3	Ivoplast	31
4.3.1	Le esigenze di Ivoplast	32
4.3.2	Sviluppo	32
4.3.3	Attività di supporto	33
5	Conclusioni	35
5.1	Analisi a consuntivo	35
5.2	Considerazioni personali	36
	Acronimi e abbreviazioni	37
	Glossario	39
	Riferimenti bibliografici	41

Elenco delle figure

1.1	Tecnologie utilizzate nella Digital Industry [18]	8
2.1	Logo Java	13
2.2	Logo Tomcat	14
2.3	Schema dei moduli del framework Synergy [17]	16
2.4	Schema di gestione delle personalizzazioni nel framework Synergy [17]	17
2.5	Logo GitLab	18
2.6	Logo Eclipse	19
2.7	Logo HeidiSQL	20
2.8	Logo Overleaf	21

Elenco delle tabelle

1.1	Obiettivi obbligatori	11
1.2	Obiettivi desiderabili	11
1.3	Obiettivi opzionali	11
5.1	Obiettivi piano di lavoro	36

Capitolo 1

Introduzione

1.1 Sanmarco Informatica S.p.A

Sanmarco Informatica è un'azienda italiana che dal 1984 si occupa di consulenza digitale e di sviluppo software. La ditta è specializzata nella progettazione e nello sviluppo di tutte quelle soluzioni mirate a riorganizzare tanto i processi aziendali quanto quelli professionali. L'azienda vanta un parco clienti complessivamente superiore alle 2000 unità, in massima parte dislocate in Italia, riuscendo anche a fornire servizi all'estero.

Ricerca e sviluppo sono la parte preminente dell'azienda, con 175 risorse dedicate e un investimento annuale superiore al 20% del fatturato [15]. Questo porta [Sanmarco Informatica \(SMI\)](#)^[gl] ad essere competitiva ed innovativa, costituendo in Italia un chiaro punto di riferimento.

Uno dei punti di forza dell'azienda è la costruzione di rapporti duraturi e costanti con i clienti, conoscerne bene le esigenze e fornire soluzioni ottimali e specifiche. Proprio questa personalizzazione software rende SMI unica sul mercato. Al centro degli obiettivi aziendali troviamo tutte le attività mirate alla completa soddisfazione del cliente, relativamente all'utilizzo delle soluzioni digitali sino al servizio pre e post vendita. L'azienda è molto presente sul territorio nazionale, ma il fulcro sono la sede centrale ed il [Centro Sviluppo \(CSV\)](#)^[gl], entrambe collocate a Grisignano di Zocco (VI). Il CSV, come dice il nome stesso, costituisce il cuore pulsante della ricerca e dello sviluppo, dove vengono progettate e sviluppate tutte le soluzioni

software utili alle svariate attività produttive che a si rivolgono a SMI [15].

1.1.1 Business Unit aziendale

L'azienda è divisa in Business unit, che d'ora in poi chiameremo **Business Unit (BU)**^[g]. Ognuna di essa lavora come un'entità autonoma, con una propria missione e alcuni obiettivi specifici. Avere al proprio interno queste Business Units autonome, permette all'azienda di articolare un piano diversificato per i propri prodotti. La BU all'interno della quale ho potuto collaborare durante il periodo di stage è quella di JMes. Essa è composta da un team leader e da una trentina di persone, comprendenti sia sviluppatori che sistemisti. Nello specifico, durante lo svolgimento dello stage, ho ricoperto il ruolo di sistemista, gestendo i progetti esterni senza sviluppare lo standard del software. La BU in cui ho collaborato ha curato tutta la parte della digital industry inerente al **Manufacturing execution system (MES)**^[g]. Le altre BU sono le seguenti:

- * JGalileo, che ha realizzato il software **Enterprise Resource Planning (ERP)**^[g] per gestire e coordinare i processi aziendali;
- * JPM, sviluppa un software di Project Management specificamente rivolto ai progetti aziendali, per facilitare il raggiungimento degli obiettivi prefissi;
- * ECM, realizza un software dedicato per la conservazione digitale a norma di legge, utile a preservare i contenuti;
- * JPA, rappresenta una soluzione software di Business Process Management: un prodotto teso a integrare e gestire i processi di diverse organizzazioni, con particolare attenzione a tutte le risorse in essere;
- * 4WORDS, è una BU che mira allo sviluppo di soluzioni web specificamente dedicate, come l'e-commerce e portali B2B.

1.1.2 Il team JMes

Come da premessa, quindi, il team JMes è relativo al software omonimo JMes, implementazione di Sanmarco Informatica del software MES. Esso è composto da

sviluppatori e sistemisti, entrambe queste figure ricoprono un ruolo fondamentale per l'avanzamento e la sostenibilità del progetto stesso.

Gli sviluppatori si occupano dello sviluppo dello standard del software, aggiornando continuamente il sistema ed eventualmente ampliandolo con nuove funzionalità utili a migliorare il software. In più, per alcuni clienti, si occupano di implementare le personalizzazioni più importanti. I sistemisti, invece, agiscono direttamente sul territorio: incontrano il cliente dove ha sede, in modo da eseguire un'analisi tecnica efficace, utile a proporre i software necessari per coprire determinate esigenze e progettando le personalizzazioni per arrivare a un programma su misura. Quando le personalizzazioni necessitano di una maggior mole di lavoro, esse vengono inviate in centro sviluppo, dove a occuparsene saranno per l'appunto gli sviluppatori.

1.1.3 Framework SCRUM

La modalità di sviluppo all'interno della BU di JMes è implementata tramite il framework SCRUM [3].

SCRUM è un framework che aiuta i team a lavorare insieme, incoraggiandoli a migliorare attraverso l'esperienza, organizzandosi in modo autonomo mentre lavorano su un problema e a riflettere sui risultati conseguiti per un miglior raggiungimento degli obiettivi.

Il framework SCRUM è utilizzato dai team di sviluppo software, ma i suoi principi possono essere applicati a tutti i tipi di lavoro in team. Viene considerato un framework Agile, SCRUM descrive in realtà una serie di riunioni, strumenti e ruoli volti ad aiutare il team a strutturare e gestire il lavoro [3].

Ruoli

Lo SCRUM team ha la caratteristica di essere autogestito, poiché contiene tutte le competenze per poter sviluppare e rilasciare software, non necessita di apporti ulteriori di team o persone esterni. Esso elenca tre ruoli distinti:

- * Product Owner: il professionista responsabile della gestione dello sviluppo del prodotto, ne definisce i requisiti interfacciandosi con il cliente. Egli si occupa di massimizzare il valore di business del progetto;

- * Scrum Master: è il membro del team la cui funzione è garantire il raggiungimento degli obiettivi fissati, inoltre ha il compito di coordinare il gruppo di sviluppatori al fine di evitare l'insorgenza di problemi o di ostacoli.
- * Developer: è il singolo sviluppatore che si occupa della creazione del progetto software.

Eventi Scrum

Per permettere un'organizzazione del lavoro ottimizzata, la tecnologia SCRUM prevede una divisione per eventi, in modo da avere un team costantemente consapevole che ha sempre un quadro complessivo del progetto. Gli eventi del framework SCRUM sono i seguenti:

- * Lo Sprinting Planning, che si colloca all'inizio dello Sprint e rappresenta un evento di pianificazione. Questa è la fase in cui vengono definite le tempistiche dello sprint e gli sviluppi da implementare;
- * Il Daily SCRUM rappresenta un evento giornaliero della durata di 15 minuti, durante il quale gli sviluppatori, di comune accordo con lo SCRUM master, correggono i loro piani in funzione dello Sprint Goal da raggiungere;
- * Lo Sprint Review è a sua volta uno degli eventi di chiusura dello sprint, costituisce per tutto il team l'occasione di ispezionare il lavoro svolto, per coglierne eventuali criticità per notare gli aspetti migliori del prodotto.
- * Per ultimo lo Sprint Retrospective, durante il quale si valutano i processi e le pratiche inerenti la collaborazione all'interno del team, al fine di migliorarne la sinergia.

Cicli di sviluppo

SCRUM è strutturato in cicli di sviluppo chiamati Sprint, essi solitamente hanno una durata compresa tra una settimana e un mese. La durata di uno sprint viene stabilita in precedenza e non può essere variata, quindi uno sprint si conclude anche se i lavori non dovessero venire ultimati.

Durante l'evento di Sprint Planning, il team pianifica l'esecuzione di alcuni task, impegnandosi a completarli entro il termine dello Sprint.

Il Daily Scrum Meeting è per il team un'occasione di confronto, per un resoconto aggiornato sull'avanzamento dei lavori e per vagliare nuove idee che magari sono state focalizzate durante la fase dello sviluppo. Alla fine di tutto, poi, si svolgeranno i due eventi di chiusura: lo Sprint Review e il Retrospective Meeting.

1.2 L'Industria 4.0

L'Industria 4.0 viene anche denominata come la quarta rivoluzione industriale. Essa rappresenta una delle ultime evoluzioni tecnologiche del settore manifatturiero, orientata all'utilizzo dell'automazione, della connettività, dell'analisi e della personalizzazione per aumentare e migliorare la produzione.

L'Industria 4.0 si basa sull'utilizzo dell'[Internet of Things \(Internet delle cose\) \(IoT\)](#)^[6], interfacciando con la rete internet oggetti fisici come per esempio le macchine di una fabbrica. Le macchine solitamente presentano moduli di sensoristica che permettono la comunicazione in tempo reale di dati e informazioni, questo permette di ottimizzare i processi produttivi e prendere decisioni in tempo reale.

L'automazione di questi processi produttivi è la base dell'Industria 4.0. L'utilizzo di sistemi automatizzati comporta un miglioramento della precisione e della velocità di produzione, riducendo gli errori ed aumentando la sicurezza sul lavoro. L'utilizzo di queste tecnologie favorisce la flessibilità della produzione, adattando i sistemi a produzioni differenti in maniera ottimizzata.

Un altro elemento fondamentale dell'Industria 4.0 è la connettività. La continua comunicazione tra macchine, personale esecutivo ed amministrativo, permette una visione completa della produzione. In questo modo è possibile individuare tempestivamente i problemi per poterli risolvere, aumentando l'efficienza dei processi produttivi.

Tutto questo rappresenta una grande opportunità per le aziende e permette di aumentare la competitività sul mercato, ai fini di offrire prodotti sempre più efficienti ad un costo "ottimizzato". Tuttavia per sfruttare queste tecnologie è

necessario che le aziende investano una parte del loro fatturato nelle tecnologie e nella formazione del personale [9].

1.2.1 Certificazione Industria 4.0

Per incentivare la digitalizzazione e la transizione all'Industria 4.0 lo stato italiano ha introdotto da un paio di anni il "Bonus Industria 4.0", questo bonus consiste in un credito d'imposta che consente alle aziende di effettuare investimenti in ottica digitalizzazione. Il bonus Industria 4.0 si rivolge a tutte le imprese residenti sul territorio nazionale. Per accedere al bonus bisogna conseguire una certificazione. La certificazione 4.0 è la dichiarazione che attesta la conformità dei beni secondo i requisiti definiti dell'Industria 4.0 che possiamo visionare negli allegati A e B del piano nazionale Transizione 4.0 [1].

La certificazione 4.0 serve per richiedere gli incentivi 4.0 così com'è scritto nell'art. 1 comma 11 della Legge 11 dicembre 2016, n. 232 nella Lgs. n 232 11 dicembre 2016:

"Per la fruizione dei benefici di cui ai commi 9 e 10, l'impresa e' tenuta a produrre una dichiarazione resa dal legale rappresentante ai sensi del testo unico delle disposizioni legislative e regolamentari in materia di documentazione amministrativa, di cui al decreto del Presidente della Repubblica 28 dicembre 2000, n. 445, ovvero, per i beni aventi ciascuno un costo di acquisizione superiore a 500.000 euro, una perizia tecnica giurata rilasciata da un ingegnere o da un perito industriale iscritti nei rispettivi albi professionali o da un ente di certificazione accreditato, attestante che il bene possiede caratteristiche tecniche tali da includerlo negli elenchi di cui all'allegato A o all'allegato B annessi alla presente legge ed e' interconnesso al sistema aziendale di gestione della produzione o alla rete di fornitura."

Nelle casistiche trattate in questa tesi, le certificazioni sono sempre state effettuate in seguito ad una perizia tecnica di un perito certificatore. L'analisi tecnica può essere suddivisa in due fasi che possono essere anche separate temporalmente:

- * fase preliminare: basata sulla verifica dei requisiti tecnici del bene fornito dal produttore;
- * fase conclusiva: verifica con il sopralluogo dell'avvenuta interconnessione del bene.

Alla fine della procedura bisogna scrivere una relazione finale dove vengono descritte le caratteristiche del bene, la modalità di interconnessione ed il modo in cui il bene è stato interconnesso al sistema informatico aziendale ai fini dell'Industria 4.0 .

In seguito alla certificazione, l'azienda è obbligata a mantenere i requisiti richiesti al fine di beneficiare dei contributi previsti dalla transizione 4.0. L'Agenzia delle entrate censisce tutti i fruitori dei contributi 4.0 e richiede report periodici sulla situazione aziendale inerente alla transizione. Inoltre prevede controlli periodici e capillari [10].

1.2.2 Manufacturing Execution System

Per interconnettere la macchina con i sistemi gestionali aziendali nasce la necessità di realizzare un software che in tempo reale prende i dati dalle macchine e li trasforma in dati utili per la gestione ed il miglioramento della produzione.

Il Manufacturing execution system (MES) è una categoria di software che viene utilizzato per la gestione ed il controllo della produzione all'interno dell'azienda. Possiamo collocarlo tra l'ERP e i sistemi di automazione della produzione ([Programmable Logic Controller \(PLC\)](#)^[8]) come possiamo vedere nella Figura 1.1.

Il MES a differenza dell'ERP, si focalizza sui processi produttivi della linea di produzione. Le principali funzionalità di un sistema MES sono le seguenti [11]:

1. Machine Data Acquisition / Collection(MDC). Acquisisce i dati da impianti diversi. Tutti i segnali provenienti dai vari impianti sono unificati e questo permette di effettuare una valutazione sulle prestazioni dei vari centri di lavoro;
2. Production Data Acquisition (PDA). Determina i dati operativi organizzativi e tecnici. Include i dati degli operatori, quelli tecnici, riportando anche gli

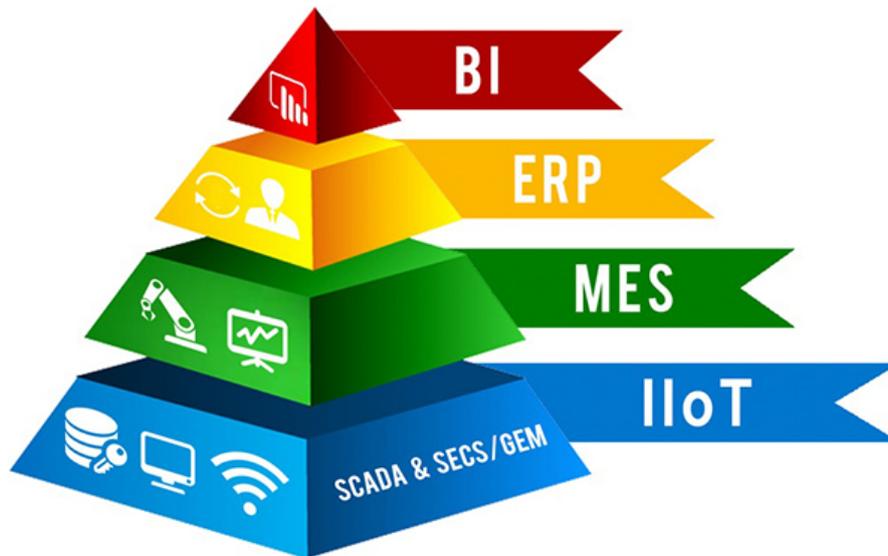


Figura 1.1: Tecnologie utilizzate nella Digital Industry [18]

stati operativi delle macchine e le quantità di produzione. In questo modo è possibile associare la produzione al personale, questo costituisce un elemento fondamentale per avere una visione unitaria dei costi;

3. Analisi delle prestazioni. Il processo di analisi delle prestazioni raccoglie i dati di produzione e prepara dei report con le informazioni principali. I lavoratori possono analizzare i report ed inserire virtualmente i dati, questo permette di avere una previsione sulla produzione in modo tale da poterla ottimizzare in termini reali;
4. Pianificazione dettagliata. A differenza di un sistema ERP che esegue una pianificazione approssimativa, il sistema MES esegue una pianificazione dettagliata. Questa pianificazione consiste in una distribuzione degli ordini su tutte le risorse disponibili, in modo preciso ed ottimale. Questo ci aiuta a monitorare la produzione permettendoci di individuare anche i colli di bottiglia;
5. Tracciabilità. Consente la tracciabilità di un prodotto, in tal modo è possibile determinare sempre la storia di un prodotto, mettendo in evidenza i materiali utilizzati ed il luogo di produzione. In questo modo vengono soddisfatti tutti i requisiti normativi riducendo gli errori;

6. Documentazione digitalizzata. Questo processo permette ai lavoratori di avere tutte le informazioni in tempo reale, riducendo l'utilizzo della carta in fabbrica. Il sistema produce ed invia tempestivamente le informazioni ai vari centri di lavoro, questo riduce la possibilità di errore perché i vari documenti non possono essere persi nelle diverse fasi di produzione.

1.2.3 Programmable Logic Controller

Il Programmable Logic Controller o PLC, è un dispositivo elettronico utilizzato per controllare le macchine industriali, monitorando i processi automatizzati.

Il PLC è un dispositivo programmabile, permettendo l'esecuzione di specifiche funzioni di automazione come la gestione dei processi industriali.

L'Industria 4.0 sta trasformando il modo in cui l'industria utilizza i PLC. Con l'avvento dell'IoT, i PLC sono diventati parte integrante dei sistemi di automazione industriale, permettendo ai produttori di monitorare e controllare i processi in tempo reale [14].

1.2.4 Gestione della memoria

La memoria di un PLC è organizzata in diverse aree, come la Program Memory, la Data Memory e la Input/Output Memory. Ognuna di queste memorie contiene informazioni specifiche che vengono utilizzate durante i vari processi.

La Program Memory contiene il codice sorgente del software di controllo del PLC. La Data Memory viene utilizzata per immagazzinare i dati che sono utilizzati dal software di controllo, come ad esempio i valori delle variabili e le letture dei sensori. Le variabili possono assumere tipi diversi come : *Integer*, *Float*, *Char*, e altri tipi. La Input/Output Memory viene utilizzata per monitorare e controllare i segnali di input/output.

Durante il corso dello stage è stato necessario utilizzare la Data Memory dei vari PLC per accedere ai dati delle macchine industriali.

1.3 Proposta di stage

1.3.1 Piano di lavoro

Sanmarco Informatica è un'azienda leader nel settore della digitalizzazione, e con questo anche nella Transizione 4.0 [15]. Lo stage in oggetto si basa sulla digitalizzazione di alcuni clienti in ottica degli incentivi Transizione 4.0. L'attività di stage si è concentrata sullo sviluppo di interfacce personalizzate per la connessione uomo-macchina. L'obiettivo principale era quello di interconnettere le macchine con JMes, un software proprietario di Sanmarco Informatica. Per permettere il collegamento viene utilizzato JMes, un software proprietario di SMI. Oltre alla parte tecnica lo stage ha avuto lo scopo di poter gestire le relazioni con i clienti, partendo dall'analisi passando per lo sviluppo del codice personalizzato su misura, fino alla messa in linea del prodotto finale.

1.3.2 Gli obiettivi

Nelle fasi preliminari allo stage, in seguito a vari incontri, sono stati definiti tutti gli obiettivi. Questo lavoro è stato fondamentale, in quanto è stato possibile definire con chiarezza i punti salienti dello stage, mettendo in chiaro le priorità. Gli obiettivi sono stati suddivisi in categorie secondo le seguenti notazioni:

- * Ob. Obiettivi obbligatori, vincolanti in quanto obiettivo primario richiesto dal committente.
- * D. Obiettivi desiderabili, non vincolanti o strettamente necessari, ma dal riconoscibile valore aggiunto.
- * Op. Obiettivi opzionali, rappresentanti valore aggiunto non strettamente competitivo.

Le sigle precedentemente indicate saranno seguite dal numero che identifica l'obiettivo. I vari obiettivi dello stage in oggetto sono raccolti nelle Tabelle: [1.1](#), [1.2](#), [1.3](#).

Obiettivi obbligatori	
Ob1	Design comunicazione tra il macchinario e software MES 2
Ob2	Implementazione dei moduli per la comunicazione tra il macchinario ed il software MES
Ob3	Documentazione sui moduli di comunicazione tra il macchinario e il MES

Tabella 1.1: Obiettivi obbligatori

Obiettivi desiderabili	
D1	Supporto tecnico per permettere al certificatore di appurare ed attestare la comunicazione in ottica transizione 4.0

Tabella 1.2: Obiettivi desiderabili

Obiettivi opzionali	
Op1	Implementazione personalizzazioni cliente
Op2	Documentazione e installazione delle nuove funzionalità

Tabella 1.3: Obiettivi opzionali

Capitolo 2

Tecnologie utilizzate

Lo scopo del capitolo è quello di e descrivere tutti gli strumenti utilizzati nel corso del progetto. Per ogni strumento verranno elencate le caratteristiche e alcuni vantaggi. Oltre ai linguaggi e il framework per lo sviluppo, verranno analizzati anche gli strumenti usati per le attività a supporto della programmazione, come ad esempio, l'[Integrated Development Environment \(IDE\)](#)^[6] per lo sviluppo.

2.1 Java

Java (logo in Figura 2.1) è un linguaggio di programmazione ad alto livello sviluppato negli anni '90. Java è diventato uno dei linguaggi più utilizzati al mondo, specialmente per la parte Java Enterprise Edition, utilizzata dalle aziende per lo sviluppo di soluzioni web [7].

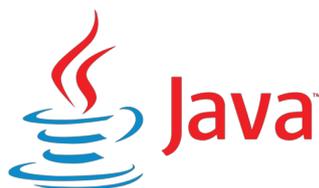


Figura 2.1: Logo Java

Durante il periodo di stage in Sanmarco Informatica, JAVA è stato utilizzato per lo sviluppo degli applicativi e delle personalizzazioni. Le caratteristiche principali

del linguaggio JAVA sono:

- * Piattaforma indipendente. Java è un linguaggio di programmazione che può essere eseguito su qualsiasi sistema operativo, questo per via della JAVA virtual machine. Questo comporta una facile distribuzione dei software scritti in JAVA;
- * Orientato agli oggetti. Java è un linguaggio orientato agli oggetti, questo vuol dire che il codice viene scritto come una serie di oggetti che comunicano tra di loro e si interfacciano. Questo rende il linguaggio modulare e flessibile;
- * Sicurezza. Java include di default diversi moduli per la sicurezza e per gestire gli errori, come per esempio la gestione automatica della memoria;
- * Scalabilità. Java è un linguaggio altamente scalabile, che permette di gestire ingenti quantità di dati mantenendo alte le prestazioni.

2.2 Tomcat

Tomcat (logo in Figura 2.2) è un web server open-source sviluppato dalla Apache Software Foundation. Si tratta di una piattaforma di servizi web che supporta Java Servlet e Java Server Pages. Tomcat rappresenta una parte integrante della piattaforma Java Enterprise Edition, offrendo diverse funzionalità per lo sviluppo web e per la gestione della sicurezza [19].



Figura 2.2: Logo Tomcat

Sanmarco Informatica predilige l'utilizzo di Tomcat per via della sua versatilità, ospitando tutti i loro servizi web sviluppati in JAVA. Per quanto riguarda le caratteristiche che rendono Tomcat uno strumento così potente e utile per lo sviluppatore troviamo:

- * Open-source. Tomcat è un software open-source, il che vuol dire che è gratuito, inoltre tutti gli sviluppatori possono contribuire ad aggiornarlo e modificarlo;
- * Scalabilità. Tomcat è altamente scalabile, permettendo di gestire grandi quantità di traffico e dati. Questo ne favorisce l'uso sia dei singoli sviluppatori che delle grandi aziende;
- * Comunità attiva. Essendo un software open-source di vasto utilizzo, esso presenta una grande comunità di sviluppatori attivi che contribuiscono al progetto supportandolo;
- * Semplicità di utilizzo. Tomcat è molto facile da installare e configurare, inoltre presenta una grande mole di informazioni online. Questo semplifica il processo di apprendimento. Inoltre può essere facilmente personalizzato, questo permette alle aziende di soddisfare tutte le varie esigenze.

2.3 Synergy

Synergy è un framework proprietario di Sanmarco Informatica. Questo framework viene utilizzato per lo sviluppo della maggior parte dei progetti di SMI. Il framework integra la parte di back-end con il front-end.

Uno dei suoi punti chiave è la gestione delle personalizzazioni, che consente a SMI di produrre soluzioni su misura con il minimo sforzo.

Il framework presenta una divisione in moduli permettendo così una gestione ottimale delle parti, vedi Figura 2.3 [17].

I moduli vengono generati automaticamente tramite un apposito [plugin](#)^[g] proprietario di SMI. In questo modo viene creata una base su cui poter lavorare. Lo schema in Figura 2.3 può essere a sua volta suddiviso in tre sottoschemi:

1. Il primo è composto dalla parte che gestisce le Api Rest e tutta la logica del software, quindi abbiamo dentro i moduli: Service, WS, Logic. Il modulo Logic viene utilizzato per l'implementazione dei vari requisiti, interagendo con il DB tramite il modulo DAO;

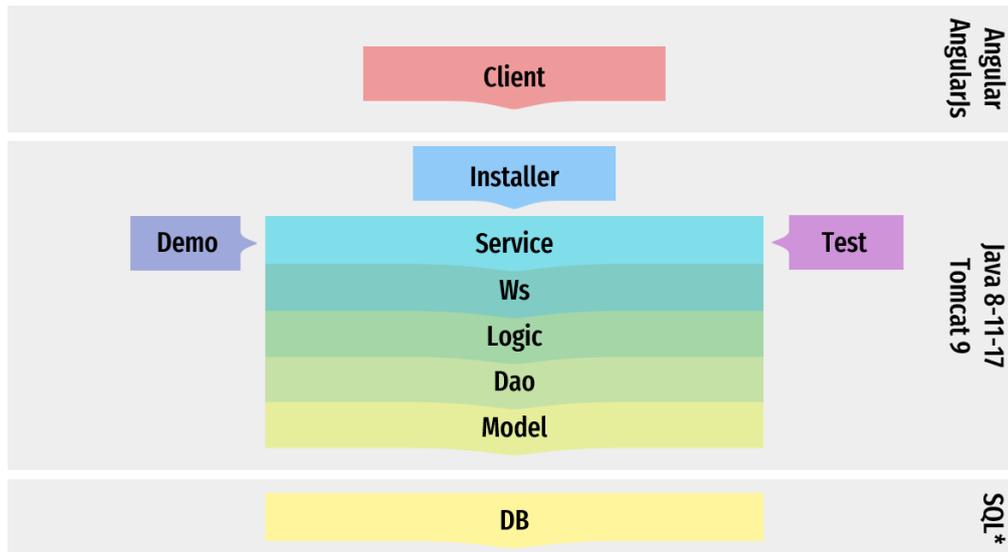


Figura 2.3: Schema dei moduli del framework Synergy [17]

- Il secondo sottoschema è composto dalla parte che gestisce i dati, quindi i moduli: DAO, Model, DB. Il modulo DAO mette a disposizione tutte le operazioni di insert, update, delete e get sulle tabelle in database; permettendo così ai moduli superiori di poter accedere ai dati salvati in database. Il modulo Model presenta una serie di classi model, che corrispondono alle tabelle presenti in database, questa gestione dà la possibilità di cambiare tipo di database senza dover effettuare modifiche al codice, ma cambiando solamente i driver di connessione;
- Il terzo sottoschema invece è composto da tutti quei moduli che possiamo chiamare esterni o di supporto. Qui sono presenti i moduli Demo e Test, che sono a supporto dello sviluppo. Il modulo Test, durante lo sviluppo dell'applicativo fornisce i fari test per i metodi Rest; invece il modulo Demo inserisce i dati di prova per facilitare lo sviluppo;

Dentro questo sottoschema è presente anche il modulo Installer, questo modulo gioca un ruolo fondamentale durante l'installazione del software dal cliente, permettendo di inserire tutti i dati di base degli applicativi; in questa fase vengono inseriti anche i dati inserenti alla licenza software.

2.3.1 Gestione delle personalizzazioni

Come detto nella Sezione 2.3, uno dei punti forti del framework Synergy è la gestione delle personalizzazioni. Il tutto è stato progettato in modo ottimale per permettere un alto livello di personalizzazione software per soddisfare tutte le necessità del cliente.

Durante lo stage è stato fatto fatto ampio uso del framework Synergy, che ha permesso di realizzare soluzioni software molto accurate e mirate a risolvere le varie problematiche.

Le personalizzazioni in Synergy seguono uno schema ben preciso e fanno uso dell'ereditarietà come si può vedere dalla Figura 2.4

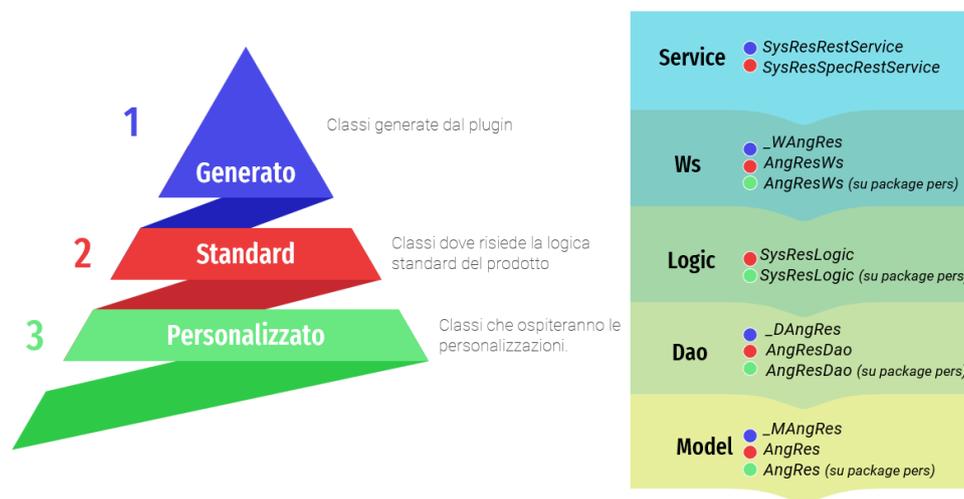


Figura 2.4: Schema di gestione delle personalizzazioni nel framework Synergy [17]

Dentro ogni progetto risiede una cartella chiamata "pers", all'interno di quella cartella è possibile creare in maniera speculare lo schema delle classi che vogliamo andare a personalizzare. In questo modo grazie all'ereditarietà il sistema prima di creare qualsiasi oggetto controlla se esiste un file personalizzato dentro la cartella "pers" con percorso speculare a quella dello standard, se questo dovesse esserci viene richiamato ed utilizzato il personalizzato, altrimenti viene utilizzato lo standard.

Questo procedimento è molto semplice da implementare e ci permette di personalizzare l'intero applicativo senza toccare minimamente lo standard o la parte autogenerata.

2.4 GitLab

Gitlab (logo in Figura 2.5) è una piattaforma di versionamento che presenta un'ampia varietà di funzionalità per la gestione del codice sorgente, la distribuzione, la gestione delle release e l'integrazione continua.

L'intera piattaforma si basa sul sistema di controllo versione denominato Git, esso rappresenta una tecnologia di sviluppo software distribuita, permettendo ai team di sviluppo di lavorare contemporaneamente su un progetto, mantenendo traccia delle modifiche effettuate e gestendo i conflitti.

Gitlab fornisce un'interfaccia completa per gestire tramite la tecnologia Git i vari progetti [4].



Figura 2.5: Logo GitLab

Durante il periodo di stage si è reso necessario l'utilizzo di GitLab per il versionamento del codice sorgente, questo permetteva una continua collaborazione tra i sistemisti ed il CSV, garantendo sempre l'integrità del software. Tra le caratteristiche e funzionalità che GitLab mette a disposizione degli sviluppatori troviamo:

- * Controllo versione con Git. L'utilizzo della tecnologia Git permette una gestione completa del codice sorgente, consentendo la collaborazione e la risoluzione dei conflitti;
- * Pipeline CI/CD*. GitLab offre una pipeline di integrazione continua. Questo consente di automatizzare le attività di sviluppo aumentando la qualità del codice sorgente;
- * Sicurezza. La piattaforma garantisce una sicurezza elevata e una gestione efficiente degli accessi tramite l'utilizzo di diversi strumenti dedicati, per garantire la sicurezza del codice sorgente;

- * Gestione degli accessi. Viene offerta una soluzione all'avanguardia per la gestione degli accessi, consentendo al team di gestire in modo efficiente le autorizzazioni date ai singoli sviluppatori per il codice sorgente.

2.5 Eclipse

Eclipse (logo in Figura 2.6) è un IDE open source, utilizzato per sviluppare software in diversi linguaggi, tra cui anche Java. Eclipse è nato nel 2001 diventando uno dei principali IDE utilizzati dagli sviluppatori [2].



Figura 2.6: Logo Eclipse

Alcune delle principali caratteristiche dell'IDE Eclipse sono:

- * Estensibilità. Eclipse è altamente personalizzabile, grazie alla presenza di plugin ed estensioni che aggiungono funzionalità al prodotto. Questa funzionalità ha permesso a Sanmarco Informatica di sviluppare una serie di plugins proprietari, utili ad implementare alcune funzionalità del framework Synergy;
- * Debugging e test. Essendo un IDE, offre una suite completa di strumenti di debugging e di testing. Questi sono di grande importanza per gli sviluppatori, perchè li aiutano ad identificare e risolvere i problemi nel codice sorgente;
- * Collaborazione. Eclipse supporta Git in modo integrato, questo permette una buona gestione delle versioni;
- * Comunità attiva. Essendo una tecnologia open-source, presenta una comunità attiva che contribuisce all'evoluzione e allo sviluppo del software.

2.6 HeidiSQL

HeidiSQL (logo in Figura 2.7) è un software per la gestione dei database. Il software presenta un'interfaccia semplice ed intuitiva, consentendo una gestione semplificata dei database. L'utilizzatore può eseguire attività come l'esecuzione di query complesse in modo semplificato [5].



Figura 2.7: Logo HeidiSQL

Durante il periodo di stage, è stato necessario gestire diversi database per clienti differenti, HeidiSQL ha semplificato questo compito. Alcune delle principali caratteristiche di HeidiSQL sono:

- * Export ed import di dati. Il sistema presenta delle funzionalità di import/export dei dati, questo permette allo sviluppatore di effettuare delle migrazioni con facilità. Inoltre è possibile esportare i dati in altri formati, tra cui CSV e SQL;
- * Backup e ripristino. È possibile effettuare backup e ripristino dei dati in modo agile, grazie alle funzionalità integrate di backup;
- * Gestione di più database. Grazie a questo strumento è possibile gestire simultaneamente più database da un'unica finestra;
- * Ottimizzazione. Con l'utilizzo di alcuni strumenti è possibile ottimizzare le prestazioni del database, tra cui l'analisi e la riparazione automatica delle tabelle.

2.7 Overleaf

Overleaf (logo in Figura 2.8) è una piattaforma di scrittura online, utilizzata solitamente per la scrittura di articoli, documenti scientifici, tesi e relazioni. Esso offre un ambiente altamente personalizzabile includendo un editor di testo LaTeX e un visualizzatore live del documento. Overleaf è stato creato anche per la condivisione dei documenti, è possibile lavorare contemporaneamente sullo stesso documento con altri utenti. Ad oggi rappresenta uno strumento indispensabile per chiunque voglia scrivere documenti scientifici [13].



Figura 2.8: Logo Overleaf

Durante lo stage, è stato necessario scrivere documentazioni per le personalizzazioni sviluppate, queste venivano inviate ai clienti come guida all'utilizzo del software. Ciò che ha contribuito a far scegliere Overleaf rispetto ad un classico elaboratore di testi sono le innumerevoli possibilità che mette a disposizione, tra cui:

- * Visualizzatore live del documento. Overleaf include un visualizzatore live, questo permette di visualizzare i cambiamenti al documento in tempo reale, in modo tale da visionare subito il risultato finale.
- * Integrazione di servizi in cloud. Tramite l'integrazione di sistemi come Dropbox e Google Drive, la piattaforma offre un sistema semplificato per la condivisione ed il backup dei documenti.
- * Collaborazione in tempo reale. Oltre alla condivisione su cloud esterni è possibile collaborare sullo stesso file direttamente dalla piattaforma, dando la possibilità a coloro che ci lavorano di essere sempre aggiornati sulle modifiche.
- * Comunità attiva di utenti. Overleaf offre una comunità di utenti attiva, pronta a risolvere i problemi, offrendo supporto continuo.

Capitolo 3

Software proprietari Sanmarco

Informatica

In questo capitolo verranno descritti i due software che sono stati al centro dello sviluppo durante il periodo di stage. Questi software sono stati integrati con personalizzazioni su misura per i clienti ed in seguito installati.

3.1 Software JMes

JMes rappresenta la soluzione software MES di Sanmarco Informatica che, attraverso gli specifici applicativi, permette di monitorare in tempo reale e non, tutte le attività svolte all'interno dei reparti produttivi. JMes permette l'avanzamento, in termini di tempi e quantità, delle fasi di lavorazione e degli ordini di produzione. L'applicazione permette di analizzare i consuntivi creando e tracciando i movimenti dei materiali, fornendo così una tracciabilità puntuale, garantendo gli standard qualitativi richiesti dal mercato [8].

JMes è composto dai seguenti applicativi:

- * JDI. Rappresenta il modulo che consente alle macchine operatrici di interfacciarsi con JMes;
- * Dialogo operatore. Ubicato all'interno dello stabilimento accanto alle macchine operatrici, rappresenta il modulo su cui dichiarare gli avanzamenti di

produzione. Viene utilizzato attraverso un browser su PC o tablet;

- * Moduli del configuratore JMes. All'interno del configuratore è possibile eseguire tutte le operazioni di configurazione delle anagrafiche e delle tabelle, le manutenzioni dei dati, la contabilizzazione delle rilevazioni, le interrogazioni e l'analisi delle statistiche, il controllo dello stato delle risorse produttive etc..

3.2 Software JDI

JDI o Java Digital Industry è un'applicativo di JMes, viene utilizzato per interconnettere la macchina con JMes.

JDI ha come obiettivo quello di aiutare l'operatore, evitandogli quindi di effettuare operazioni manuali, come la dichiarazione dei pezzi prodotti da una singola macchina oppure la segnalazione di uno stato di produzione. L'applicativo JDI manda tutti i dati a JMes e per questomotivo non prevede un'interfaccia grafica da fornire all'operatore in stabilimento. JDI presenta comunque un'interfaccia grafica che permette la configurazione di tutti i parametri di configurazione con JMes.

Al suo interno, sono sviluppati una serie di [driver](#)^[8] per la connessione con i PLC. I driver implementano dei protocolli noti e permettono uno scambio dati continuo, consentendo a JMes di essere sempre aggiornato sugli stati della macchina e sulla lavorazione.

Di seguito una lista con degli esempi di protocolli supportati da JDI:

- * Modbus TCP/IP;
- * OPC UA;
- * MQTT;
- * Siemens S7;
- * Heidenhain V1 e V2.

Durante il corso dello stage, sono state implementate e sviluppate modifiche e personalizzazioni per la connessione di macchine operative con protocollo Siemens S7 e OPC UA.

3.3 Interfaccia JMes - JDI

Si può considerare che, durante una lavorazione in fabbrica, un operatore dovrebbe effettuare due operazioni differenti: svolgere operazioni sulla macchina per far avanzare il processo produttivo e successivamente fare un consuntivo sul sistema MES. Si può portare come esempio la produzione di sedie in plastica tramite una pressa idraulica; in questo caso una volta completato il processo di produzione, l'operaio dovrà dichiarare quante sedie sono state prodotte, quanti scarti ci sono stati e per quanto tempo la macchina ha lavorato oppure è stata ferma.

JDI aiuta l'operatore interfacciandosi direttamente a JMes, in questo modo l'operatore non dovrà dichiarare volta per volta ogni informazione, ma dovrà solamente confermare i dati che verranno comunicati direttamente a JMes. Questa soluzione permette di ottimizzare il lavoro, riducendo al minimo gli errori.

La connessione fisica tra JDI e JMes avviene tramite le interfacce di configurazione di entrambi i software. Ogni applicativo conosce l'URL su cui risponde l'altro, in questo modo possono comunicare in maniera continua. I due software vengono configurati inserendo al loro interno l'anagrafica delle macchine per avere una visione coerente e precisa dei dati.

Capitolo 4

Analisi e sviluppo: Interfacce e Personalizzazioni

In questo capitolo, verranno trattate le attività di sviluppo e personalizzazione effettuate per diversi clienti SMI durante il periodo di stage. Il focus è lo sviluppo delle personalizzazioni utili a soddisfare esigenze specifiche, inoltre verrà trattato il supporto offerto con continuità ed in costante contatto, nel ruolo di sistemista ricoperto in azienda durante lo stage.

4.1 Solmec

Solmec è un'azienda leader a livello mondiale nella progettazione e produzione di macchine per l'edilizia [16]. L'azienda offre una vasta gamma di macchine per la movimentazione terra, quali escavatori, pale gommate e macchine per la demolizione. La gamma di prodotti è in continua evoluzione perché la Solmec investe costantemente in ricerca e sviluppo. Un'industria con questa fisionomia necessita costantemente di aggiornamenti nonché di nuovi applicativi software, utili a massimizzare processi e risultati.

4.1.1 Le esigenze di Solmec

Durante la prima parte del periodo di stage, è stato possibile visitare l'azienda, per capirne le esigenze e dotarla di soluzioni utili per ottenere i requisiti indispensabili per la certificazione "Industria 4.0". L'obiettivo di Solmec era connettere con l'applicativo JMes un centro di fresatura a più assi. La connessione con il MES era indispensabile per una gestione più accurata e precisa delle attività e, non ultimo, per il conseguimento della utile certificazione precedentemente citata. Il software JMes gestisce la produzione in quattro fasi separate:

1. Inizio attrezzaggio. L'operatore attrezza la macchina in base alle indicazioni fornite da JMes: dal caricamento del materiale alla gestione di eventuali strumenti specifici, in modo da rendere la macchina pronta alla fase di lavoro;
2. Fine attrezzaggio. L'operatore dichiara su JMes che la fase di attrezzaggio è stata completata, quindi la macchina è pronta per la fase di lavoro;
3. Inizio lavoro. L'operatore addetto, successivamente all'avvio manuale della macchina tramite il di controllo, dichiara su JMes l'inizio della fase di lavoro;
4. Fine lavoro. A operazione conclusa l'operatore dichiara di aver completato la fase di lavoro, confermando al contempo le dichiarazioni sulla quantità dei pezzi prodotti nonché degli scarti conseguenti.

L'azienda Solmec, successivamente alla presentazione del software e delle sue caratteristiche, sviluppate nello standard JMes, richiede alcune personalizzazioni volte ad automatizzare il passaggio da inizio attrezzaggio sino a inizio lavoro. In pratica, vuole risparmiare all'operatore di dichiarare manualmente l'inizio lavoro, in modo da evitarne l'erronea omissione su JMes.

4.1.2 Sviluppo

In seguito alle richieste del cliente, si è personalizzato il software JMes in modo che successivamente alla fase di inizio attrezzaggio, la fase di inizio lavoro si sarebbe avviata automaticamente, dopo l'avvio della macchina da parte dell'operatore. Per sviluppare questa funzionalità è stata utilizzata la variabile di stato della macchina.

Questa variabile indica in tempo reale lo stato della macchina, evidenziando la fase in cui si trova la macchina: lavoro, pausa o stop.

Lo sviluppo si basa sulla personalizzazione di due classi JAVA, standard di JMes:

- * *JDiServiceManager*;
- * *MesJdiLogic*.

La classe *JDiServiceManager* riceve i vari dati in arrivo da JDI scatenando delle azioni. In questo caso viene effettuato l'override del metodo di rielaborazione delle azioni in modo da personalizzarlo. All'interno del metodo viene intercettato il messaggio di " Inizio lavoro " che viene lanciato dopo che l'operatore ha fatto partire la macchina. Intercettato il messaggio di start della macchina, viene chiamato il metodo *startAutomaticWork* della classe *MesJdiLogic*. Questo metodo è stato creato appositamente per realizzare la personalizzazione in modo da richiamare le funzioni che permettono l'inizio della fase di lavoro su JMes. Praticamente, i due sistemi dialogano continuamente tra loro. Quando l'operatore fa partire il lavoro sulla macchina, JMes in automatico avvia la fase di lavoro: l'operatore ha un passaggio in meno su JMes, quindi meno possibilità di errore.

Problemi riscontrati

Durante la fase di configurazione dei software JMes e JDI, è stato notato un malfunzionamento della variabile conta pezzi. Tale variabile contatore di JMes, dovrebbe essere incrementata ogni volta che la variabile *boolean* di fine lavoro del PLC viene valorizzata a *True*. Come precedentemente spiegato, l'applicativo JDI è responsabile del collegamento tra PLC e JMes.

Invece, successivamente all'installazione e configurazione dei software, la variabile risultava incrementata di un'unità ad ogni apertura delle porte del centro di lavoro di fresatura. Poiché tale apertura è prevista in diverse fasi quali pulizia, controllo o modifiche, gli incrementi risultavano sensibilmente superiori al dato reale dei pezzi prodotti. Controlli e verifiche effettuati in sede presso il cliente hanno rilevato come il problema fosse nel driver di connessione tra JDI e il PLC Heidenhain. Il driver di JDI leggeva in maniera errata la memoria del PLC: le variabili venivano

lette partendo da una erronea posizione di memoria. La correzione ha richiesto un aggiornamento sul driver, svolto da alcuni sistemisti presenti in CSV. Aggiornato il driver è stato configurato il sistema del cliente con i nuovi aggiornamenti, risolvendo il problema.

4.2 Mimest

Una seconda esperienza ha interessato l'azienda Mimest, azienda in grande crescita che dal 2005 è diventata l'azienda di riferimento italiano per la produzione di semilavorati metallici con la tecnica del Metal Injection Molding [12]. Unendo la tecnologia dello stampaggio ad iniezione dei materiali polimerici termoplastici con la sinterizzazione delle polveri metalliche.

Mimest serve una grande varietà di settori: a partire dal settore medicale, moda, lusso, difesa, occhialeria e meccanica di precisione in generale.

4.2.1 Le esigenze di Mimest

L'azienda presentava la necessità di certificare con l'Industria 4.0 un forno a nastro Cremer. All'interno del forno passavano dei carrelli con dentro i vari pezzi di varie leghe, stampati precedentemente da alcune presse o stampanti 3D. L'azienda necessitava una personalizzazione della gestione delle ricette, in quanto all'interno del forno a nastro continuo, transitavano vari pezzi inerenti a produzioni differenti. Pertanto era necessario effettuare una gestione delle ricette in parallelo.

4.2.2 Sviluppo

Per implementare la soluzione software all'esigenza sopra citata, è stata fatta un'analisi della memoria del forno Cremer, da cui è risultata la presenza di un *array* di 10 posizioni che conteneva le variabili per la gestione delle ricette in parallelo. Quindi sul PLC del forno Cremer era possibile gestire 10 ricette in parallelo.

La prima parte dell'implementazione è stata configurare JDI per effettuare una lettura continua delle variabili del PLC del forno Cremer. Una volta configurate

tutte le variabili su JDI è stato possibile avere accesso all'intera area di memoria interessata per effettuare lo sviluppo.

La seconda parte invece si è basata sull'implementazione su JMes della classe JAVA *WriteToJDIche*, che gestiva la scrittura delle variabili su JDI e quindi a sua volta sul PLC del forno Cremer.

Una volta completata questa funzionalità è stato possibile effettuare l'inizio attrezzaggio del forno su JMes, in maniera parallela con più ricette. La ricetta consiste nella bolla di esecuzione della macchina, che in pratica indica all'operatore il pezzo da produrre. Prima di effettuare l'inizio attrezzaggio JMes controllava il numero di ricette già in lavorazione. Nel caso in cui fosse stata uguale a 10, non avrebbe permesso il caricamento della nuova ricetta. Bisognava quindi attendere il completamento di una delle ricette in lavorazione.

4.2.3 Attività di supporto

Durante le visite dal cliente, abbiamo fornito costante supporto anche per quanto riguarda la messa in rete del forno Cremer. Il forno non era connesso ad internet in quanto passasse per un [firewall](#)^[g] inserito precedentemente all'interno del quadro della macchina. In seguito a vari test ci siamo accorti che all'indirizzo IP che doveva avere il forno, sulla porta 80, rispondeva il pannello di controllo del firewall, permettendoci di individuarlo e di segnalarlo all'azienda che lo ha prontamente rimosso per permettere la corretta configurazione dei software JDI e JMes.

4.3 Ivoplast

Un altro intervento ha interessato Ivoplast, altra azienda leader, stavolta nel settore della produzione di soluzioni plastiche avanzate per abitazione e industria [6]. Ivoplast presenta un team di ingegneri altamente qualificati che lavora costantemente al miglioramento dei propri prodotti.

4.3.1 Le esigenze di Ivoplast

Durante la fase di analisi, il cliente ha evidenziato l'esigenza di recuperare l'ultima ricetta per ogni singola macchina, ove il processo di produzione ne richiedesse la necessità. La ricetta, come nella Sezione 4.2.2, consiste nella bolla di esecuzione della macchina, che in pratica indica all'operatore il pezzo da produrre. Le personalizzazioni sono state mirate a implementare la funzionalità "Riprendi lavoro". Di fatto questa si rende necessaria perché le presse idrauliche che stampano i prodotti possono necessitare di manutenzione durante una fase di lavoro. L'esigenza cui far fronte, era poter interrompere la fase di lavoro al fine di svolgere le eventuali attività di manutenzione nonché, ovviamente, continuare la produzione dal punto in cui era stata interrotta: qui interviene, e risolve, l'implementazione di "Riprendi lavoro".

4.3.2 Sviluppo

Per sviluppare questa soluzione è stato necessario intervenire sia sulla parte logica che sulla parte grafica, inserendo dei nuovi bottoni all'interno dell'interfaccia "Dialogo Operatore". Selezionando una delle varie macchine è adesso possibile cliccare sul bottone "Riprendi lavoro". Per quanto riguarda la parte logica, queste funzionalità sono state implementate all'interno di una classe JAVA chiamata *WorkRestoreUtil*. All'interno di questa classe, è stata sviluppata una funzione che gestisce il recupero dal database delle informazioni inerenti la ricetta. Inizialmente, viene recuperato l'ultimo evento eseguito sulla macchina interessata, registrato nella tabella *MesEvt*. Questa tabella contiene tutti gli eventi svolti e in svolgimento sulle varie macchine. Nel caso in cui non venisse trovato alcun evento per la macchina interessata, si passa ad un controllo sulla tabella *MesEvtBck*, nella quale vengono salvati tutti gli eventi già saldati. Recuperato l'evento occorre ricercare dettagli della ricetta. Queste informazioni sono rilevabili dalla tabella *MesEvtDet*. Nel caso in cui, invece, l'evento fosse stato recuperato dalla tabella *MesEvtBck*, i dettagli verrebbero ricavati dalla tabella *MesEvtDetBck*. Acquisiti i dettagli della ricetta, occorre richiamare dallo standard JMes la funzionalità "Inizio lavoro", direttamente con i campi precompilati. Ciò permette di continuare nuovamente il lavoro dal punto in cui era stato interrotto.

Nel caso in cui non si dovesse trovare alcun evento con relativa ricetta su *MesEvt* o su *MesEvtBck*, un errore aviserebbe dell'assenza di eventi.

4.3.3 Attività di supporto

Durante il secondo mese di stage, gran parte del tempo è stata dedicata a seguire tutte le attività del cliente, dalla fase di analisi sino alla messa in linea in produzione del prodotto. Successivamente alla fase di sviluppo, abbiamo effettuato, in un ambiente dedicato, alcuni test funzionali. Uno di questi test prevedeva la simulazione di una ripresa dell'utilizzo delle macchine in seguito ad un'interruzione per lavori di manutenzione. Sono stati eseguiti due test separati in modo da controllare il corretto recupero degli eventi dalla tabella *MesEvt* e *MesEvtBck*. I test hanno simulato fedelmente i comportamenti dell'utente in fabbrica, permettendo una corretta valutazione delle personalizzazioni sviluppate. In conseguenza di questi e dell'approvazione dell'IT Manager dell'azienda Ivoplast, si è infine effettuato l'upload della soluzione nell'ambiente di produzione.

Capitolo 5

Conclusioni

5.1 Analisi a consuntivo

Le 320 ore di stage svolte presso Sanmarco Informatica hanno seguito il piano di lavoro considerando i vari obiettivi, come da piano.

La previsione sul numero di ore dedicato alle attività si è rivelata corretta. Possono essere riassunte le ore impiegate per lo svolgimento delle attività:

- * Formazione sulle tecnologie: 80 ore. Per questa attività è stato fondamentale il supporto del team JMes in CSV. Il team mi ha aiutato a comprendere le nuove tecnologie grazie a diversi corsi, permettendomi così di sviluppare sul loro framework synergy e di comprendere a pieno i software Jmes e JDI;
- * Contatti con il cliente ed i fornitori: 120 ore. Durante il periodo di stage, ricoprendo un ruolo da sistemista, è stato fondamentale coltivare i contatti con i cliente ed i fornitori, questi mi permettevano di capire le esigenze per poi ottimizzare gli sviluppi;
- * Analisi ed implementazione di nuove funzionalità: 120 ore. Per tutti i clienti che ho trattato, è stata effettuata un'analisi dei requisiti ed una successiva implementazione delle nuove funzionalità. È stata fondamentale la presenza del tutor che mi ha seguito durante i vari passaggi.

I vari obiettivi sono stati portati a termine. La Tabella [5.1](#) riassume gli obiettivi e il loro livello di raggiungimento.

Obiettivo	Tipologia	Stato
Ob1 - Design comunicazione tra macchinario e software MES	Obbligatorio	Completato
Ob2 - Implementazione dei moduli per la comunicazione tra il macchinario e il software MES	Obbligatorio	Completato
Ob3 - Documentazione sui moduli di comunicazione tra il macchinario ed il software MES	Obbligatorio	Completato
D1 - Supporto tecnico per permettere al certificatore di appurare ed attestare la comunicazione in ottica transizione 4.0	Desiderabile	Completato
Op1 - Implementazione personalizzazioni cliente	Opzionale	Completato
Op2 - Documentazione e installazione delle nuove funzionalità	Opzionale	Completato

Tabella 5.1: Obiettivi piano di lavoro

5.2 Considerazioni personali

Lo stage svolto presso Sanmarco Informatica è stato molto formativo. Sono stato inserito in un buon team che mi ha sempre aiutato e supportato durante tutto il periodo di stage.

Ho scelto questo stage da sistemista perché mi dava la possibilità di ricoprire un ruolo diverso da quelli che ho ricoperto durante le altre mie esperienze lavorative in altre aziende. Questo mi ha permesso di imparare a relazionarmi bene con i clienti, in quanto il sistemista rappresenta la prima figura a cui i clienti si rivolgono. Grazie a questo stage ho avuto la possibilità di conoscere diverse realtà industriali, questo ha contribuito ad arricchire il mio bagaglio culturale, mostrandomi l'applicazione finale dei software prodotti, rendendo tangibili i risultati ottenuti.

Ho in buona parte testato la metodologia di sviluppo Agile SCRUM, rimanendo molto soddisfatto, questa metodologia era stata usata anche per il progetto di Ingegneria del Software.

Devo infine ringraziare Sanmarco Informatica, il team JMes in centro sviluppo, il mio tutor ed il team di sistemisti per avermi permesso di vivere un'esperienza che ha sicuramente arricchito il mio curriculum, insegnandomi l'importanza della collaborazione all'interno dell'ambiente lavorativo.

Acronimi e abbreviazioni

BU [Busines Unit.](#) 2

CSV [Centro Sviluppo.](#) 1

ERP [Enterprise resource planning.](#) 2, 39

IDE [Integrated Development Environment.](#) 13, 39

IoT [Internet of Things \(Internet delle cose\).](#) 5, 39

MES [Manufacturing execution system.](#) 2, 39

PLC [Programmable Logic Controller.](#) 7, 39

SMI [Sanmarco Informatica.](#) 1

Glossario

IoT L'Internet delle cose , in inglese Internet of Things (IoT) fa riferimento all'estensione di internet al mondo degli oggetti, che acquisiscono una propria identità digitale in modo da poter comunicare con altri oggetti nella rete e poter fornire servizi agli utenti. [37](#)

Driver Un driver è uno specifico applicativo software che permette la comunicazione con un dispositivo hardware. [24](#)

ERP l'Enterprise Resource Planning è un tipo di software che le organizzazioni utilizzano per gestire le attività quotidiane di business, come ad esempio contabilità, procurement, project management e gestione del rischio. [37](#)

Firewall Un firewall, è un dispositivo di difesa perimetrale di una rete, che può anche svolgere funzioni di collegamento tra due o più segmenti di rete, garantendo quindi la sicurezza della rete. [31](#)

IDE Un Integrated Development Environment è un software progettato per la realizzazione di applicazioni che aggrega strumenti di sviluppo, in un'unica interfaccia utente grafica. [37](#)

MES Il Manufacturing Execution System è un software che controlla e gestisce la produzione di un'azienda in modo automatizzato e integrato. [37](#)

PLC Programmable Logic Controller è un dispositivo digitale industriale programmabile, specializzato nella gestione dei processi industriali. Esso elabora i segnali provenienti dai sensori presenti in un impianto industriale. [37](#)

Plugin Un plugin è un programma non autonomo che interagisce con un altro programma per ampliarne o estenderne le funzionalità originarie. [15](#)

Le definizioni presenti nel glossario sono state prese da Wikipedia [\[20\]](#).

Riferimenti bibliografici

Siti web consultati

- [1] *Certificazione per l'Insustria 4.0*. URL: <https://www.orimos.dev/industria-4-0/cose-la-certificazione-4-0/> (cit. a p. 6).
- [2] *Eclipse*. URL: [https://it.wikipedia.org/wiki/Eclipse_\(informatica\)](https://it.wikipedia.org/wiki/Eclipse_(informatica)) (cit. a p. 19).
- [3] *Framework SCRUM*. URL: <https://www.atlassian.com/it/agile/scrum> (cit. a p. 3).
- [4] *Gitlab*. URL: <https://it.wikipedia.org/wiki/GitLab> (cit. a p. 18).
- [5] *HeidiSQL*. URL: <https://en.wikipedia.org/wiki/HeidiSQL> (cit. a p. 20).
- [6] *Ivoplast*. URL: <https://www.ivoplast.com/it/> (cit. a p. 31).
- [7] *JAVA*. URL: [https://it.wikipedia.org/wiki/Java_\(linguaggio_di_programmazione\)](https://it.wikipedia.org/wiki/Java_(linguaggio_di_programmazione)) (cit. a p. 13).
- [8] *JMes*. URL: <https://www.sanmarcoinformatica.com/mes-manufacturing-execution-system/> (cit. a p. 23).
- [9] *L'Industria 4.0*. URL: <https://www.economyup.it/innovazione/cos-e-1-industria-40-e-perche-e-importante-saperla-affrontare/> (cit. a p. 6).
- [10] *Mantenimento requisiti Industria 4.0*. URL: <https://www.galvanotecnica.org/it/importantissimo-%5C%E2%5C%80%5C%93-obbligo-mantenimento-dei-requisiti-4-0> (cit. a p. 7).

- [11] *Manufacturing execution system (MES)*. URL: <https://forcam.com/en/mes-importance-and-benefits-manufacturing-execution-system/> (cit. a p. 7).
- [12] *Mimest*. URL: <https://www.mimest.tech/ita/> (cit. a p. 30).
- [13] *Overleaf*. URL: <https://en.wikipedia.org/wiki/Overleaf> (cit. a p. 21).
- [14] *Programmable Logic Controller (PLC)*. URL: https://it.wikipedia.org/wiki/Controllore_logico_programmabile (cit. a p. 9).
- [15] *Sanmarco Informatica*. URL: <https://www.sanmarcoinformatica.com/> (cit. alle pp. 1, 2, 10).
- [16] *Solmec*. URL: <https://www.solmec.it/it/> (cit. a p. 27).
- [17] *Synergy - Documentazione interna Sanmarco Informatica* (cit. alle pp. 15–17).
- [18] *Tecnologie utilizzate nella Digital Industry*. URL: <https://store.cmcts.com.vn/mes-nha-may-thong-minh> (cit. a p. 8).
- [19] *Tomcat*. URL: https://it.wikipedia.org/wiki/Apache_Tomcat (cit. a p. 14).
- [20] *Wikipedia*. URL: <https://it.wikipedia.org> (cit. a p. 40).