

# Indice

<b>Elenco delle figure</b>	<b>iii</b>
<b>Introduzione</b>	<b>1</b>
<b>1 Azienda e Requisiti</b>	<b>3</b>
1.1 L'azienda . . . . .	3
1.1.1 Le attività dell'azienda . . . . .	4
1.2 Analisi dei requisiti . . . . .	7
1.2.1 Introduzione . . . . .	7
1.2.2 Requisiti Funzionali . . . . .	8
1.2.3 Requisiti Non Funzionali . . . . .	21
1.2.4 Requisiti Di Dominio . . . . .	22
<b>2 Strumenti e tecnologie</b>	<b>23</b>
2.1 Tecnologie utilizzate . . . . .	23
2.1.1 Web Application - Angular . . . . .	23
2.1.2 Linguaggi utilizzati . . . . .	24
2.1.3 Database - Microsoft SQL Server . . . . .	27
2.2 Strumenti utilizzati . . . . .	32
2.2.1 Microsoft Entity Framework Core . . . . .	32
<b>3 Implementazione</b>	<b>34</b>
3.1 Modelli . . . . .	34
3.1.1 AnalisiChimiche . . . . .	34
3.1.2 Anomalie . . . . .	35
3.1.3 Colate . . . . .	36
3.1.4 Correzioni . . . . .	37
3.1.5 Schede Output . . . . .	38
3.1.6 Output Colata . . . . .	38

3.1.7	Rapporti Lavoro . . . . .	40
3.2	Controller . . . . .	40
3.2.1	AnalisiChimicheController . . . . .	41
3.2.2	ColateController . . . . .	42
3.2.3	OuptutColataController . . . . .	42
3.2.4	RapportiDiLavoroController . . . . .	43
3.2.5	SchedeOutputController . . . . .	46
3.2.6	ShiftController . . . . .	47
3.2.7	UserController . . . . .	47
<b>4</b>	<b>Interfaccia</b>	<b>49</b>
4.1	Login . . . . .	49
4.2	Storico Colate . . . . .	49
4.3	Schede Colaticci . . . . .	49
4.4	Rapporti Di Lavoro Forno Fusorio . . . . .	49
4.5	Rapporti Di Lavoro Forni a Bacino . . . . .	49
4.6	Rapporti Di Lavoro Colata Continua . . . . .	49
4.7	Rapporti Di Lavoro Magazzino Pani . . . . .	49
4.8	Analisi Chimiche . . . . .	49
4.9	Materie e Elementi Chimici . . . . .	49
4.10	Gestione Dizionari . . . . .	49
4.11	Secuirity . . . . .	49
<b>5</b>	<b>Sviluppi futuri e conclusione</b>	<b>50</b>
5.1	Sviluppi futuri . . . . .	50
5.1.1	Analisi Metallografiche . . . . .	50
5.1.2	Gestione carichi magazzino . . . . .	50
5.2	Conclusioni . . . . .	50

# Elenco delle figure

# Introduzione

Nel corso della storia il mondo ha assistito a diverse rivoluzioni industriali. La prima, avvenuta intorno al 18° secolo in Gran Bretagna, ha portato all'incremento della produzione con il passaggio dalla lavorazione manuale a quella tramite macchine. Infatti si è passati da un sistema agricolo-artigianale-commerciale a un sistema industriale moderno con l'utilizzo di macchine azionate da energia meccanica e altre nuove forme energetiche quali i combustibili fossili. Questa trasformazione avvenne soprattutto nei settori tessile e metallurgico con l'introduzione della macchina a vapore. La seconda, avvenuta circa un secolo dopo, ha introdotto le catene di montaggio e l'utilizzo di nuove fonti di energia quali energia elettrica e petrolio. La terza invece, iniziata a metà del 20° secolo, ha integrato i computer e le telecomunicazioni avanzate nei processi della produzione industriale. Durante la Terza rivoluzione industriale infatti sono stati integrati anche i PLC (programmable logic controller) nei macchinari per automatizzare processi e raccogliere e condividere i dati di produzione. E ora si è arrivati alla quarta rivoluzione industriale, indicata anche come *Industry 4.0*.

L'*Industry 4.0* è incentrata sul concetto di *smart factory*, che si compone di tre parti: *smart production*, ovvero la creazione di collaborazione tra tutti gli elementi presenti nella produzione (operatore, macchine e strumenti), *smart service*, ovvero tutte le infrastrutture che permettono di integrare tutti i sistemi, e *smart energy*, ovvero la creazione di sistemi performanti con sprechi di energia ridotti e con un occhio di riguardo ai consumi energetici.

L'obiettivo dello stage è stato quindi quello dell'inserimento nel team di sviluppo della società *Adipso S.r.l.* per la progettazione e lo sviluppo dell'applicativo *Rapporti di Lavoro*, una web application per la gestione automatica della produzione di leghe in alluminio in colata continua partendo da una materia prima riciclata. In particolare questa web application consentirà il monitoraggio e l'interazione dell'operatore con gli impianti automatici di produzione, proprio per andare incontro alle esigenze della normativa *Industry 4.0*.

Una particolare attenzione sarà dedicata al design delle interfacce utente che dovranno essere semplici e intuitive, per ridurre al minimo gli errori degli operatori e ottimizzare l'efficienza operativa.

Questa tesi è organizzata come segue: il primo capitolo contiene una panoramica generale dell'azienda *Adipso S.r.l* presso la quale è stato svolto lo stage, sulle richieste del cliente e sulle funzionalità previste dalla web application. Il secondo capitolo presenta l'architettura della web application e le tecnologie utilizzate. Il terzo capitolo descrive in modo generale la struttura della web application e dei suoi componenti. Il quarto capitolo contiene un breve manuale utente. Infine, l'ultimo capitolo contiene le considerazioni finali sullo sviluppo della web application e eventuali sviluppi futuri.

# Capitolo 1

## Azienda e Requisiti

### 1.1 L'azienda

Come già anticipato nel capitolo precedente, lo stage è stato svolto presso l'azienda Adipso S.r.l..

Adipso S.r.l. nasce dalla società Adding S.r.l. la quale, fin dal 2004 si pone l'obiettivo di inserirsi nel mondo degli Integratori di sistemi di automazione e gestione di produzione industriale e fornire a questo settore tutta la propria competenza, esperienza e professionalità in ambito di sviluppo e progettazione software in ambito industriale.

Adipso propone soluzioni tecniche complete aggiornate ai più recenti strumenti dell'informatica ma con particolare attenzione alla stabilità, alla scalabilità e alla robustezza del sistema. Infatti, il punto di forza della società è l'essere un team di professionisti aggiornati alle più recenti tecnologie informatiche e sistemistiche e con esperienza nell'ambito dell'automazione industriale, della gestione della produzione e dello sviluppo software di integrazione tra sistemi e macchine.

L'obiettivo è quindi quello del successo dei progetti dei clienti e di implementare la migliore soluzione in base alle esigenze tecniche, gestionali, economiche e strategiche dei clienti.

Le aree applicative di Adipso variano dalle industrie farmaceutiche a quelle chimiche, dal settore metallurgico a quello dei prodotti personali e da società di ingegneria a quelle di produzione di materie plastiche.

### 1.1.1 Le attività dell'azienda

All'interno dei prossimi paragrafi verranno descritte brevemente le principali attività svolte dall'azienda, realizzate sempre con un occhio di riguardo alle esigenze del cliente.

#### Progettazione

Per quanto riguarda l'attività di progettazione, Adipso si occupa di progettare e realizzare impianti elettro-pneumatici e quadri elettrici di potenza e di automazione, oltre che alla fornitura della strumentazione di campo e alla stesura dei documenti di progetto, quali la Functional Specification (Specificazione Funzionale, FS), l'Hardware Design Specification (Specificazione di disegno Hardware, HDS), la Software Design Specification (Specificazione di disegno Software, SDS) e il manuale operatore generale per il sistema, oltre che a tutti i relativi allegati.

#### Automazione

Per quanto riguarda l'automazione invece, Adipso si occupa di sviluppare sistemi software di automazione e ingegneria per quadri elettrici, o sistemi software di supervisione. Inoltre si occupa dell'integrazione o del revamping (ammodernamento) di sistemi esistenti e della loro manutenzione. Le principali tecnologie usate in questo ambito sono i **PLC** (programmable logic controller), gli storicizzatori come Proficy Historian, e gli **SCADA** (Supervisory Control And Data Acquisition).

**PLC** I PLC solitamente vengono utilizzati in industrie per la gestione o per il controllo dei processi industriali. I compiti svolti da un PLC quindi vanno dal realizzare semplici sequenze di operazioni al controllo delle movimentazioni alla realizzazione di sofisticati sistemi di controllo distribuiti, che prevedono più PLC che collaborano tra di loro. La struttura dei PLC è simile a quella dei normali PC. Sono quindi composti da un alimentatore, una CPU, da memorie interne o esterne, come RAM e ROM. La più grande differenza con i PC normali consiste nel fatto che i PLC dispongono anche di schede di ingresso e uscita digitali e, all'occorrenza, analogiche, utilizzate per il controllo di grandezze. Inoltre i PLC possono disporre anche di schede di comunicazione per comunicare con computer o con altri PLC.

Per funzionare al meglio i PLC devono essere programmati. La programmazione di un PLC avviene con un PC sul quale un software specializzato permette di

creare programmi per la gestione industriale che verranno poi caricati sulla CPU del PLC.

**SCADA** Supervisory Control And Data Acquisition, o SCADA, indica un sistema informatico distribuito per il monitoraggio e la supervisione di sistemi fisici. Solitamente uno SCADA è un software installato su normali computer o server che permette il funzionamento e la gestione di sistemi di supervisione senza necessariamente scrivere codice tramite linguaggi di programmazione, punto di maggiore importanza dato che coloro che realizzano i sistemi SCADA spesso sono tecnici e non informatici o programmatori. Gli SCADA generalmente sono utilizzati nelle centrali di controllo di fabbriche, aeroporti o grandi complessi di edifici.

Gli SCADA vengono utilizzati come interfaccia verso operatori o altri sistemi. Le più importanti funzioni di uno SCADA sono l'acquisizione dei dati mediante opportuni driver di comunicazione verso gli apparati in campo, tra i quali i già citati PLC, a loro volta connessi a sensori o attuatori, la rappresentazione del dato sullo schermo di un PC o di uno smartphone, la storicizzazione del dato stesso su archivi locali o distribuiti, la gestione degli allarmi, che consente di visualizzare eventuali problemi all'impianto e eventualmente di avvisare il responsabile con una chiamata, una mail o più generalmente con il suono di una sirena per avvisare tutto il reparto dell'anomalia. L'ultima funzionalità e forse la più richiesta nell'ultimo periodo è quella dell'interazione con sistemi di livello superiore, ovvero fare in modo che lo SCADA faccia da interfaccia tra il sistema fisico e sistemi di livello superiore come un sistema MES o una web application che consente la gestione dell'impianto con più facilità.

### Web Application

Lo sviluppo di Web Application è l'attività principale svolta durante il project work e consiste nello sviluppo di applicazioni web dedicate a partire dalle specifiche esigenze del cliente, definite durante la fase dell'analisi dei requisiti. Le principali tecnologie utilizzate per lo sviluppo di web application sono HTML 5, Angular 8, .NET Core, e altre. Queste tecnologie verranno descritte nel dettaglio nei prossimi capitoli.

### Database

Lo sviluppo di Web Application in parte consiste anche nella relazionizzazione di database per la storicizzazione dei dati. Nell'attività relativa ai database



vengono studiate le problematiche di gestione dei dati nei sistemi informativi e proposte soluzioni le quali tengano conto delle tecniche di progetto e delle tecnologie di gestione dei dati. I principali database utilizzati sono Microsoft SQL Server, utilizzato durante il project work e che verrà descritto nel dettaglio nei prossimi capitoli, e Oracle.

### **Trend e Report**

Nello sviluppo di applicativi software o web application, a volte nelle richieste del cliente è compresa anche la possibilità di visualizzare e/o stampare reportistica contenente i dati storicizzati. Per questo Adipso si occupa anche della parte relativa a trend e report, ovvero dell'analisi dei dati e della generazione di reportistica manuale e/o automatizzata in base alle richieste dei clienti. I principali strumenti di reportistica utilizzati sono Crystal Reports e Microsoft Reporting Service.

### **Convalida**

L'ultima attività è quella di convalida. La convalida viene spesso richiesta dai clienti, soprattutto quelli che operano in ambito farmaceutico, e consiste nella redazione, revisione e approvazione della documentazione di progetto, secondo il ciclo di vita conforme alla normativa GAMP5, descritta di seguito. I documenti in questione comprendono non solo quelli descritti nell'attività di progettazione ma anche altri, come il piano dei test, che comprende la descrizione dell'ambiente di test, la strategia e la metodologia dei test, i test di modulo eseguiti per gli oggetti standard utilizzati sia a livello PLC che a livello SCADA e i verbali di rilascio e i certificati di installazione hardware e software, che certifica tutto quello che è stato installato

**GAMP** Ormai nell'ambito sanitario è richiesta sempre più spesso la convalida dei sistemi informatici. Questa convalida consiste nell'attestazione delle capacità di un sistema di funzionare efficacemente e in modo riproducibile fornendo prestazioni e risultati conformi a standard predefiniti. Tutto questo viene conseguito tramite la produzione di evidenze oggettive. La Good Automated Manufactured Practice (GAMP) fornisce un aiuto nell'ambito della convalida dei sistemi informatici. Gli obiettivi della guida GAMP sono quelli di facilitare l'interpretazione dei requisiti normativi, stabilire un linguaggio e una terminologia comune e promuovere un approccio del ciclo di vita del sistema basato sulle buone pratiche. GAMP non è una vera e propria norma, ma fornisce una guida

che, se applicata con buon senso, diventa un buon aiuto per la convalida. Il metodo GAMP si basa su diversi concetti chiave, tra i quali ci sono la conoscenza del prodotto e dei processi e il coinvolgimento dei fornitori.

## 1.2 Analisi dei requisiti

### 1.2.1 Introduzione

Un requisito è una descrizione dei servizi che un sistema software deve fornire e i vincoli da rispettare, sia in fase di sviluppo che durante la fase di operatività del software stesso. L'analisi dei requisiti è la prima fase del processo di sviluppo di un software, in quanto ha lo scopo di definire le funzionalità che il prodotto finale deve offrire. L'analisi dei requisiti quindi non è altro che il processo di ricerca, analisi, documentazione e verifica dei servizi richiesti dal cliente e dei vincoli entro i quali i servizi stessi devono operare.

I requisiti si dividono in requisiti utente e requisiti di sistema. I requisiti utente sono la descrizione in linguaggio naturale, con eventuale aggiunta di diagrammi e tabelle, dei servizi che il sistema deve fornire e dei vincoli operativi. Solitamente i requisiti utente sono scritti per e con il cliente. Invece i requisiti di sistema sono specificati mediante la stesura di un documento strutturato che descrive in modo dettagliato i servizi che il sistema software deve fornire. Infatti definiscono cosa deve essere implementato e può essere usato come base per il progetto. Un'altra possibile classificazione divide i requisiti in requisiti funzionali, non funzionali e di dominio. Questi requisiti verranno descritti nei prossimi paragrafi, insieme ai requisiti della web application *Rapporti di Lavoro*. Durante la fase di analisi dei requisiti possono sorgere diverse problematiche, come requisiti che possono essere interpretati in modo differente (Ambiguità), requisiti che non includono la descrizione di tutte le caratteristiche richieste (Incompletezza) e conflitti o contraddizioni nella descrizione delle caratteristiche del sistema (Inconsistenza). Spesso questi problemi non sono risolvibili dato che, per esempio, è impossibile produrre un documento che contenta tutti i requisiti nella loro completezza, e spesso i requisiti vanno in conflitto tra di loro. Nei paragrafi successivi verranno descritti i requisiti della web application *Rapporti di Lavoro*, che viene utilizzata come applicazione di gestione e visualizzazione dei rapporti di lavoro di un impianto dedicato alla produzione di leghe in alluminio in colata continua, partendo da una materia prima riciclata, che viene introdotta nel forno fusorio, nel quale viene sciolta e lavorata. Terminata la lavorazione nel forno fusorio, questa colata viene sversata in modo alternato in

due forni a bacino, nei quali vengono eseguite altre lavorazioni, come la pulizia del materiale e la correzione con altri materiali correttivi. Dopodichè la colata viene sversata da entrambi i forni a bacino in modo alternato nel reparto di colata continua, nel quale vengono creati dei pani, ovvero dei blocchi costituiti dal materiale in lavorazione. Una volta terminata la lavorazione di questi pani, essi vengono conservati nel reparto Magazzino Pani.

### 1.2.2 Requisiti Funzionali

I requisiti funzionali descrivono le funzionalità e i servizi del sistema. Rispondono quindi alla domanda: *cosa deve essere fatto?*. Di seguito vengono riportati i requisiti funzionali della web application *Rapporti di Lavoro*.

#### Login

La funzionalità di login permette all'utente di autenticarsi utilizzando le proprie credenziali di dominio. Infatti la pagina di login consente di inserire nome utente e password. Una volta inseriti dall'utente, il sistema verifica la correttezza delle credenziali inserite confrontandole con le credenziali di dominio dell'utente stesso. Se i dati inseriti risultano corretti, l'utente viene autenticato e può utilizzare la web application, altrimenti viene segnalato all'utente l'errore e viene mostrata di nuovo la schermata di login. Questa fase di login può essere bypassata se l'utente avvia la web application tramite il software di supervisione SCADA, dato che l'utente che utilizza il software di supervisione ha già effettuato il login.

#### Visualizzazione dello Storico Colate

La funzionalità di visualizzazione dello storico colate consente di visualizzare l'elenco delle colate registrate nel sistema, sia quelle base, ovvero quelle relative al forno fusorio, sia quelle specifiche, ovvero quelle relative agli altri reparti dell'impianto.

La pagina di visualizzazione dell'elenco delle colate sarà divisa in due sezioni orizzontali, la prima contenente l'elenco delle colate base, la seconda contenente l'elenco delle colate specifiche. Per entrambe le colate vengono visualizzate le seguenti informazioni:

- Numero della colata;
- Data e ora di inizio della colata;

- Data e ora in cui la colata è dichiarata come pronta;
- Data e ora di fine della colata;
- Operatore che ha dichiarato la colata come pronta;
- Operatore che ha dichiarato la colata come conclusa;
- Stato della colata;
- Quantità di O<sub>2</sub> consumata;
- Quantità di CH<sub>4</sub> consumata.

Oltre alle informazioni elencate in precedenza, per le colate specifiche vengono visualizzate anche le seguenti informazioni:

- Numero della colata progressivo;
- Destinazione della colata, ovvero verso quale forno a bacino viene indirizzata la colata;
- Peso totale della colata.

All'interno di questa pagina è inoltre presente un filtro che consente di visualizzare solo le colate all'interno di un determinato range temporale.

### Visualizzazione delle informazioni delle colate

Per ogni colata è possibile visualizzare le relative informazioni dettagliate, oltre a quelle già elencate nel paragrafo precedente. Le informazioni in questione vengono descritte di seguito:

**Lega di riferimento** Per quanto riguarda la lega di riferimento vengono visualizzate informazioni sulla lega che deve essere prodotta dalla colata di riferimento, tra cui il nome, il cliente e eventuali note.

**Output della colata** Per quanto riguarda gli output della colata vengono visualizzate le informazioni sugli output prodotti dalla colata di riferimento. Gli output della colata saranno approfonditi successivamente.

**Materiali e materie prime** Per quanto riguarda i materiali e le materie prime viene visualizzato un elenco dei vari materiali che sono stati caricati nel corso della colata per mantenere gli standard produttivi e, per ogni materiale, la materia prima di riferimento, con relative informazioni come codice, nome, descrizione e la percentuale di metallo.

**Visualizzazione delle Schede Colaticci attuali**

La funzionalità di visualizzazione delle schede colaticci attuali consente di visualizzare l'elenco di queste schede attualmente aperte e le relative informazioni. Le schede colaticci sono l'equivalente virtuale di un contenitore che contiene tutti gli output derivanti dalle colate in corso. E' presente una scheda colaticci per ogni contenitore, e, solitamente, un contenitore per ogni forno. Anche in questo caso, come per le colate, è possibile visualizzare sia le schede relative al forno fusorio, quindi alle colate base, che quelle relative ai forni a bacino, quindi alle colate specifiche.

Le informazioni visualizzate sono le seguenti:

- Numero della scheda;
- Data di apertura della scheda;
- Operatore che ha aperto la scheda;
- Contenitore di riferimento;
- Peso totale degli output presenti nel contenitore, in tonnellate;
- Elenco degli output contenuti nel contenitore di riferimento, con possibilità di aggiunta, modifica e spostamento.

A ogni output sono collegate le seguenti informazioni:

- Numero colata che ha generato l'output di riferimento;
- Lega di riferimento;
- Tipo di output;
- Quantità totale generata, in tonnellate;
- Data di inizio della colata relativa;
- Data di dichiarazione dell'output;
- Data di scorifica, ovvero la data nella quale sono state rimosse le scorie dalla lega in produzione;
- Settimana relativa, calcolata in base alla data di scorifica;
- Qualità dell'output;
- Operatore che ha inserito l'output;

- Provenienza dell'output;
- Numero di casse prodotte;
- Eventuali Note.

In ogni scheda è possibile aggiungere nuovi output con le informazioni descritte sopra, modificare la quantità e la data di scorporo per ogni output e spostare un output in un'altra scheda relativa allo stesso forno. Inoltre queste schede possono essere chiuse. Una volta chiusa una scheda per un determinato contenitore, viene automaticamente creata una nuova scheda per lo stesso contenitore.

### **Visualizzazione delle Schede Colaticci chiuse**

La funzionalità di visualizzazione delle schede colaticci chiuse consente di visualizzare l'elenco di queste schede chiuse e le relative informazioni. Anche in questo caso, come per le schede attuali, è possibile visualizzare sia le schede relative al forno fusorio, quindi alle colate base, che quelle relative ai forni a bacino, quindi alle colate specifiche.

Le schede possono essere filtrate temporalmente per visualizzare solo quelle relative a un determinato range temporale, oppure può essere cercata una singola scheda con il relativo numero.

Le informazioni visualizzate per le schede chiuse sono le seguenti:

- Numero della scheda;
- Contenitore di riferimento
- Stato della scheda, che può essere spedito, se la scheda è stata chiusa, o campionato, se la scheda è stata chiusa e sono stati inseriti i dati di spedizione;
- Progressivo del camion utilizzato per spedire la scheda;
- Il numero del documento di trasporto, o DDT;
- Data di apertura della scheda;
- Operatore che ha aperto la scheda;
- Data di spedizione della scheda;
- Operatore che ha spedito la scheda;

- Data di associazione della scheda, ovvero data di inserimento delle informazioni di spedizione;
- Operatore che ha associato la scheda;
- Peso totale degli output presenti nel contenitore, in tonnellate;
- Elenco degli output contenuti nel contenitore di riferimento, con possibilità di modifica e spostamento.

Sarà possibile quindi, oltre che a modificare e spostare ogni singolo output, associare il numero progressivo del camion e relativo numero di DDT a ogni scheda.

### **Visualizzazione dei Rapporti di Lavoro del Forno Fusorio**

La funzionalità di visualizzazione dei rapporti di lavoro del forno fusorio consente di visualizzare le informazioni relative alle operazioni effettuate nel reparto forno fusorio. Le informazioni sono organizzate per turno e di default vengono visualizzati i dati inseriti dall'inizio del turno precedente fino ad adesso, ma la data iniziale può essere modificata per visualizzare le informazioni più datate. Tutti i dati visualizzati in questa pagina quindi sono filtrati per range temporale.

All'interno del rapporto di lavoro del forno fusorio vengono visualizzate le seguenti informazioni:

- Informazioni sul turno attuale, come data di inizio, data di fine e numero del turno;
- Informazioni sulla colata in corso, come numero della colata, lega di riferimento e data e ora di inizio

Oltre a queste informazioni vengono visualizzati anche degli elenchi, descritti successivamente. Per ogni elenco è possibile visualizzare i dati e aggiungerne di nuovi.

**Elenco carichi materiale** In questo elenco vengono visualizzati i materiali che devono essere caricati nella colata e tutte le cariche effettuate nel range temporale selezionato, divise per materiale. Una carica può essere aggiunta, rimossa e eliminata. Inoltre è possibile modificare la percentuale di organico di una singola carica o di più cariche in blocco.

**Elenco output colata** In questo elenco vengono visualizzate le registrazioni degli output collegati alla colata in corso effettuate nel range temporale selezionato. Le informazioni visualizzate sono le stesse elencate nella descrizione delle schede colaticci. E' possibile aggiungere un output colata direttamente dal rapporto di lavoro.

**Elenco fermi** In questo elenco vengono visualizzati i fermi impianto che si sono verificati nel range temporale selezionato e tutti quei fermi che non sono stati giustificati. E' possibile filtrare i fermi per fermi giustificati o non giustificati, oppure è possibile visualizzarli tutti. Le informazioni visualizzate per i fermi sono le seguenti:

- Data e ora inizio e fine;
- Durata;
- Descrizione dei cinque perchè e dell'azione.

La giustificazione di un fermo avviene tramite una struttura detta dei cinque perchè. *Cinque Perchè* è un metodo che consente di esplorare le relazioni causa-effetto per un problema ponendosi una semplice domanda. Lo scopo di questo metodo è quello di determinare le cause del difetto. In particolare, nei rapporti di lavoro viene utilizzato per stabilire le motivazioni di un fermo e, in aggiunta, viene anche richiesta l'azione correttiva per la risoluzione delle problematiche che hanno portato al fermo impianto.

**Elenco Anomalie** In questo elenco vengono visualizzate le anomalie che si sono verificate nel range temporale selezionato. Le informazioni visualizzate sono le seguenti:

- Data;
- Operatore;
- Tipo Anomalia;
- Descrizione dell'anomalia.

**Elenco note turno** Questo elenco è diviso in due parti: l'elenco delle note relative al turno corrente e quelle relative al turno precedente. Negli elenchi sono visualizzate le note inserite nel turno corrente e quelle inserite in un range temporale che va dalla data di inizio selezionata e la fine del turno precedente. Le informazioni visualizzate sono le seguenti:



- Data;
- Operatore;
- Descrizione.

### Visualizzazione dei Rapporti di Lavoro dei Forni a Bacino

La funzionalità di visualizzazione dei rapporti di lavoro dei forni a bacino consente di visualizzare le informazioni relative alle operazioni effettuate nel reparto dei forni a bacino. Le informazioni sono organizzate per numero di colata specifica e di default vengono visualizzati i dati relativi alla colata in corso e al primo forno bacino. ma la colata può essere selezionata da un menù a tendina che contiene tutte le colate ordinate dalla più recente alla più datata e il forno può essere selezionato tra tutti i forni a bacino presenti in impianto. Tutti i dati visualizzati in questa pagina quindi sono filtrati in base al numero colata selezionato.

All'interno del rapporto di lavoro dei forni a bacino vengono visualizzate le seguenti informazioni:

- Informazioni sul turno attuale, come data di inizio, data di fine e numero del turno;
- Informazioni sulla colata selezionata, come numero della colata, lega di riferimento e data e ora di inizio. Nel caso in cui la colata fosse chiusa vengono visualizzate anche la data e l'ora di fine.

Oltre a queste informazioni vengono visualizzati anche degli elenchi, descritti successivamente. Per ogni elenco è possibile visualizzare i dati e aggiungerne di nuovi.

**Elenco carichi materiale** In questo elenco vengono visualizzati i materiali che devono essere caricati nella colata e tutte le cariche effettuate divise per materiale. I materiali da caricare corrispondono ai correttivi selezionati nella fase di calcolo di correzione delle analisi chimiche. Quando una proposta di correzione viene accettata, questo elenco viene compilato con i materiali da caricare e la relativa quantità. Una carica può essere aggiunta, rimossa e eliminata. Inoltre è possibile modificare la percentuale di organico di una singola carica o di più cariche in blocco. Oltre ai materiali pianificati, possono essere aggiunti nuovi materiali non pianificati, con quantità da caricare prevista uguale a 0.

**Elenco output colata** In questo elenco vengono visualizzate le registrazione degli output collegati alla colata selezionata. Le informazioni visualizzate sono le stesse elencate nella descrizione delle schede colaticci. E' possibile aggiungere un output colata direttamente dal rapporto di lavoro.

**Elenco Anomalie** In questo elenco vengono visualizzate le anomalie che si sono verificate durante la lavorazione della colata selezionata. Le informazioni visualizzate sono le seguenti:

- Data;
- Operatore;
- Tipo Anomalia;
- Descrizione dell'anomalia.

**Elenco note turno** Questo elenco è diviso in due parti: l'elenco delle note relative al turno corrente e quelle relative al turno precedente. Negli elenchi sono visualizzate le note inserite nel turno corrente e quelle inserite in un range temporale che va dalla data di inizio alla data di fine del turno precedente. Le informazioni visualizzate sono le seguenti:

- Data;
- Operatore;
- Descrizione.

### **Visualizzazione dei Rapporti di Lavoro del reparto Colata Continua**

La funzionalità di visualizzazione dei rapporti di lavoro della colata continua consente di visualizzare le informazioni relative alle operazioni effettuate nel reparto di colata continua. Le informazioni sono organizzate per numero di colata e di default vengono visualizzati i dati relativi all'ultima colata specifica che è stata chiusa, ma la colata può essere selezionata da un menù a tendina a che contiene tutte le colate chiuse ordinata dalla più recente alla più datata. Tutti i dati visualizzati in questa pagina quindi sono filtrati in base al numero colata selezionato.

All'interno del rapporto di lavoro della colata continua vengono visualizzate le seguenti informazioni:

- Informazioni sul turno attuale, come data di inizio, data di fine e numero del turno;
- Informazioni sulla colata selezionata, come numero della colata, lega di riferimento, data e ora di inizio e di fine e durata totale;
- Temperatura attuale del bacino;
- Grafico con storico della temperatura del crogiolo;
- Tempi di funzionamento dei macchinari in impianto.

Oltre a queste informazioni vengono visualizzati anche degli elenchi, descritti successivamente. Per ogni elenco è possibile visualizzare i dati e aggiungerne di nuovi. Ogni elemento di questi elenchi avrà un colore diverso in base alla macchina di riferimento, in modo da identificare subito la macchina.

**Elenco Rallentamenti** In questo elenco vengono visualizzati i rallentamenti che si sono verificati durante la lavorazione della colata selezionata. Le informazioni visualizzate sono le seguenti:

- Macchina sulla quale si è verificato il rallentamento;
- Data di inizio e di fine;
- Operatore;
- Causa del rallentamento.

**Elenco Anomalie** In questo elenco vengono visualizzate le anomalie che si sono verificate durante la lavorazione della colata selezionata. Le informazioni visualizzate sono le seguenti:

- Macchina sulla quale si è verificata l'anomalia;
- Data;
- Operatore;
- Tipo Anomalia;
- Descrizione dell'anomalia.

**Elenco fermi** In questo elenco vengono visualizzati i fermi impianto che si sono verificati durante la lavorazione della colata selezionata e tutti quei fermi che non sono stati giustificati. E' possibile filtrare i fermi per fermi giustificati o non giustificati, oppure è possibile visualizzarli tutti. Le informazioni visualizzate per i fermi sono le seguenti:

- Macchina sulla quale si è verificato il fermo;
- Data e ora inizio e fine;
- Durata;
- Descrizione dei cinque perchè e dell'azione.

La giustificazione di un fermo avviene tramite una struttura detta dei cinque perchè.

**Elenco note** Questo elenco è diviso in due parti: l'elenco delle note relative alla colata selezionata e quelle relative alle dieci colate precedenti. Negli elenchi sono visualizzate le note inserite durante la lavorazione della colata corrente e quelle inserite durante la lavorazione delle dieci colate precedenti a quella selezionata. Le informazioni visualizzate sono le seguenti:

- Numero della colata;
- Data;
- Operatore;
- Descrizione.

### Visualizzazione dei Rapporti di Lavoro del reparto Magazzino Pani

La funzionalità di visualizzazione dei rapporti di lavoro del magazzino pani consente di visualizzare le informazioni relative alle operazioni effettuate nel reparto magazzino pani. Le informazioni sono organizzate per numero di colata e di default vengono visualizzati i dati relativi all'ultima colata specifica che è stata chiusa, ma la colata può essere selezionata da un menù a tendina a che contiene tutte le colate chiuse ordinata dalla più recente alla più datata. Tutti i dati visualizzati in questa pagina quindi sono filtrati in base al numero colata selezionato.

All'interno del rapporto di lavoro della colata continua vengono visualizzate le seguenti informazioni:

- Informazioni sul turno attuale, come data di inizio, data di fine e numero del turno;
- Informazioni sulla colata selezionata, come numero della colata, lega di riferimento, data e ora di inizio e di fine e durata totale.

Oltre a queste informazioni vengono visualizzati anche degli elenchi, descritti successivamente. Per ogni elenco è possibile visualizzare i dati e aggiungerne di nuovi.

**Elenco Pesatura Pacchi** In questo elenco vengono visualizzate le quantità dei pani che arrivano dal reparto di colata continua, suddivisa per pani corti e lunghi e per calcolo automatico e manuale, con anche il totale numero di pani immagazzinati.

**Elenco fermi** In questo elenco vengono visualizzati i fermi impianto che si sono verificati durante la lavorazione della colata selezionata e tutti quei fermi che non sono stati giustificati. E' possibile filtrare i fermi per fermi giustificati o non giustificati, oppure è possibile visualizzarli tutti. Le informazioni visualizzate per i fermi sono le seguenti:

- Macchina sulla quale si è verificato il fermo;
- Data e ora inizio e fine;
- Durata;
- Descrizione dei cinque perchè e dell'azione.

La giustificazione di un fermo avviene tramite una struttura detta dei cinque perchè.

**Elenco Anomalie** In questo elenco vengono visualizzate le anomalie che si sono verificate durante la lavorazione della colata selezionata. Le informazioni visualizzate sono le seguenti:

- Data;
- Operatore;
- Tipo Anomalia;
- Descrizione dell'anomalia.

**Elenco note turno** Questo elenco è diviso in due parti: l'elenco delle note relative al turno corrente e quelle relative al turno precedente. Negli elenchi sono visualizzate le note inserite nel turno corrente e quelle inserite in un range temporale che va dalla data di inizio alla data di fine del turno precedente. Le informazioni visualizzate sono le seguenti:

- Data;
- Operatore;
- Descrizione.

### Visualizzazione delle Analisi Chimiche e calcolo delle correzioni

Questa funzionalità permette di visualizzare l'elenco delle analisi chimiche che vengono svolte per il forno fusorio e per i forni a bacino. Le analisi chimiche vengono utilizzate per ottenere la composizione della colata analizzata. La composizione ottenuta consiste nell'elenco degli elementi chimici presenti, il relativo peso e la percentuale, calcolata in base al peso totale della colata. Questo elenco consente di verificare se la colata è all'interno degli standard produttivi previsti. Per ogni forno vengono visualizzate le informazioni sulla colata attuale, come lega, numero colata, numero colata progressiva e il peso. E' inoltre possibile visualizzare un barcode che identifica il numero della colata, in modo da poterlo scansionare. Per ogni analisi chimica vengono visualizzate le seguenti informazioni:

- Id Analisi progressivo;
- Tipo di analisi chimica;
- Data in cui l'analisi chimica è stata effettuata;
- Identificativo del quantometro, ovvero dello strumento utilizzato per ottenere i risultati delle analisi chimiche.

Inoltre, vengono visualizzati anche i seguenti elenchi:

**Dettaglio Analisi** Elenco che contiene il nome dell'elemento, il peso e la percentuale relativa rilevati e i range di percentuali in cui ogni elemento deve rientrare per rispettare gli standard produttivi previsti dalla normativa e dal cliente.

**Elenco Proposte** Per ogni analisi chimica possono essere calcolate le proposte di correzione. In questo elenco vengono visualizzate tutte le proposte possibile, con id progressivo e correttivo di riferimento.

**Dettaglio Proposta** Vengono visualizzati i dettagli delle proposte già eventualmente accettate o quelle nuove da accettare e rifiutare, con l'id del materiale correttivo, la descrizione e la quantità da caricare in chilogrammi. Per ogni proposta è possibile eliminare un correttivo, aggiungerne uno nuovo e modificare la quantità da caricare di ogni correttivo.

### **Visualizzazione e gestione delle anagrafiche delle materie prime e degli elementi chimici**

Questa funzionalità permette di visualizzare le anagrafiche, ovvero le informazioni, relative alle materie prime utilizzate nell'impianto e agli elementi chimici. Oltre a visualizzare le informazioni è possibile aggiungere nuove anagrafiche, sia per le materie prime che per gli elementi chimici, oltre che modificare quelle già esistenti. Inoltre per le materie prime che vengono utilizzate come correttivo, ovvero quelle materie prime che vengono aggiunte alla colata in corso per correggere i risultati delle analisi chimiche per rispettare gli standard produttivi, è possibile aggiungere o modificare l'associazione della materia stessa con gli elementi chimici che la compongono, con l'indicazione della percentuale e se la materia prima è un correttivo per un determinato elemento chimico oppure no.

### **Visualizzazione, aggiunta e modifica delle anagrafiche generali dell'applicazione**

Questa funzionalità permette di visualizzare le anagrafiche, ovvero le informazioni, relative ai tanti aspetti dell'applicativo. I dizionari permettono un più facile inserimento dei dati da parte degli operatori, fornendo una lista di possibili valori in quei campi non liberamente compilabili ma che possono assumere solo determinati valori. Oltre a visualizzare le informazioni è possibile aggiungere nuove anagrafiche e modificare quelle già esistenti. Tra i vari dizionari possiamo trovare:

- L'elenco dei perchè, utilizzati per la giustificazione dei fermi;
- L'elenco degli operatori che hanno accesso all'applicativo;
- L'elenco dei gruppi operatori, utilizzati per l'assegnamento delle funzionalità a ogni utente;

- L'elenco delle cause dei rallentamenti;
- L'elenco dei tipi di anomalia.

### Gestione delle security

Ogni singolo utente appartiene a uno dei gruppi elencati nel *Dizionario Gruppi Utente* citato in precedenza. Ogni gruppo utente ha accesso a parti diverse della web application e a diverse funzionalità. L'accesso a diverse aree e funzionalità viene gestito dall'amministratore tramite la pagina di gestione delle security. Per ogni gruppo utente deve essere indicato a quali aree e a quali funzionalità critiche può accedere.

### Logout

La funzionalità di logout permette all'utente di uscire dall'applicazione. Una volta effettuato il logout l'utente viene scollegato dall'applicazione e il sistema mostra di nuovo la pagina di login.

## 1.2.3 Requisiti Non Funzionali

I requisiti non funzionali descrivono le modalità operative e di gestione del sistema. Definiscono quindi vincoli sullo sviluppo del sistema stesso. Di seguito vengono riportati i requisiti non funzionali della web application *Rapporti di Lavoro*.

### Durata della sessione

La sessione si interrompe ogni ora, con il sistema che esegue in automatico il logout e visualizza la pagina di login. Il parametro della durata della sessione è configurabile in base alle necessità.

### Storicizzazione dei dati

Ogni informazione visualizzata dal sistema è memorizzata all'interno di un database, che contiene quindi tutti i dizionari, tutte le note, tutte le rilevazioni e tutti i dati che sono stati citati nel paragrafo precedente.

### Aspetti sulla sicurezza

Le password degli operatori vengono memorizzate nel database. Per una questione di sicurezza queste password devono essere criptate.



### 1.2.4 Requisiti Di Dominio

I requisiti di dominio sono requisiti derivati dal dominio applicativo del sistema software o da necessità dettate dagli utenti. Di seguito vengono riportati i requisiti funzionali della web application *Rapporti di Lavoro*.

#### **Accesso consentito a personale autorizzato**

L'accesso alla web application può essere eseguito solo da personale autorizzato, che ha un proprio utente nella rete dell'impianto e che ha i permessi per accedere ai software di supervisione.

## Capitolo 2

# Strumenti e tecnologie

All'interno di questo capitolo verranno descritte le tecnologie e gli strumenti utilizzati per lo sviluppo della web application *Rapporti di Lavoro*.

### 2.1 Tecnologie utilizzate

#### 2.1.1 Web Application - Angular

##### Web Application

Una *Web Application* è un'applicazione distribuita, ovvero un'applicazione accessibile tramite il web. In particolare una web application può essere resa accessibile su *internet*, e quindi resa accessibile a tutti, oppure all'interno di una *intranet*<sup>1</sup>, e quindi accessibile solo all'interno di un sistema informatico. Per la realizzazione dell'applicativo *Rapporti di Lavoro* è stato deciso di sviluppare una web application, in modo da renderla accessibile semplicemente con l'utilizzo di un browser installato su un semplice computer o un tablet.

---

<sup>1</sup>Una rete aziendale privata e isolata dalla rete esterna, che offre servizi accessibili solo dall'interno della rete aziendale.

## Angular

Per la realizzazione della web application *Rapporti di Lavoro* si è deciso di utilizzare *Angular*. *Angular* è un framework opensource<sup>2</sup> dedicato appunto allo sviluppo di web application e sviluppato principalmente da Google. L'intenzione originaria era quella di creare uno strumento semplice e rapido per lo sviluppo di applicazioni in grado di girare su qualunque piattaforma e facilmente fruibili da qualsiasi tipo di dispositivo, come pc desktop, tablet o smartphone. Il più grande punto di forza di Angular è che le applicazioni vengono eseguite interamente dal browser. In questo modo anche le applicazioni più complesse e sofisticate risultano comunque veloci, leggere e facilmente fruibili.

Quando si parla di Angular non si può non parlare di *Single Page Application* o *SPA*, ovvero applicazioni web il quale utilizzo avviene tramite una singola pagina web. Questo garantisce una user experience molto più fluida e dinamica, comparabile a quella di un'applicazione desktop.

Angular non solo può essere utilizzato per le web application, ma si dimostra efficace anche in applicazioni mobile e/o desktop (tramite strumenti esterni come Ionic e Electron). Inoltre sfrutta anche moltissime librerie per l'integrazione con ulteriori servizi terzi con estrema facilità come Google o Amazon Web Services.

### 2.1.2 Linguaggi utilizzati

Nel paragrafo seguente verranno descritti i linguaggi utilizzati per la realizzazione della web application *Rapporti di Lavoro*.

## HTML

*HTML*, abbreviazione di *Hypertext Markup Language*, è lo standard usato per formattare e impaginare documenti ipertestuali<sup>3</sup> ed è un linguaggio di markup. Un linguaggio di markup è un insieme di regole che descrivono i meccanismi di rappresentazione di un testo. Il linguaggio HTML consente quindi di descrivere semanticamente la struttura di un documento web attraverso tag. HTML rappresenta dunque la struttura portante delle pagine web: su questa struttura si possono aggiungere modifiche grafiche, grazie ai fogli di stile CSS, che verranno descritti successivamente, ed elementi dinamici. L'HTML è un linguaggio di

---

<sup>2</sup>Un software è definito opensource se il codice sorgente viene reso disponibile per la modifica, lo studio e l'utilizzo.

<sup>3</sup>Un ipertesto è un insieme di documenti correlati tra di loro tramite parole chiave.

pubblico dominio e la sua sintassi è stabilita dal *W3C* (*World Wide Web Consortium*), che si occupa di favorire lo sviluppo di tutte le potenzialità del World Wide Web. Il funzionamento del linguaggio HTML è quindi quello di gestire i contenuti associandone la struttura grafica grazie all'utilizzo di tag diversi. Infatti ogni tag identifica un diverso ruolo dei contenuti.

Un documento HTML inizia con una dichiarazione del tipo di documento, utilizzata per dare al browser l'indicazione per interpretare e visualizzare il documento stesso. Dopo la dichiarazione del tipo di documento, questo presenta una struttura ad albero annidata composta da sezioni delimitate da tag che a loro volta contengono altre sezioni sempre delimitate da tag. La struttura più esterna del documento è compresa tra i tag `<html>` e `</html>`. All'interno di questo tag lo standard prevede la definizione di due sottosezioni distinte: la sezione di intestazione (o *header*), delimitata tra i tag `<head>` e `</head>`, che contiene informazioni di controllo solitamente non visualizzate dal browser come collegamenti verso file esterni (file CSS o di script), e la sezione del corpo (o *body*), delimitata tra i tag `<body>` e `</body>`, che contiene la parte informativa, ovvero il testo, le immagini e tutto ciò che costituisce la parte che deve essere visualizzata dal browser. Oltre all'indicazione sulla struttura del documento lo standard prevede che le sezioni non debbano essere sovrapposte, ovvero che ogni sezione deve essere chiusa prima di iniziare la sezione successiva.

## CSS

*CSS*, abbreviazione di *Cascading Style Sheets*, è il linguaggio usato per definire la formattazione dei documenti *HTML*. Così come per l'*HTML*, le regole per comporre il *CSS* sono contenute in un insieme di direttive emanate dal *W3C*. L'introduzione del *CSS* è stata utile per separare i contenuti delle pagine HTML dalla loro formattazione. Questa struttura garantisce anche il riutilizzo del codice, ovvero consente di richiamare parti di codice già scritte senza la necessità di doverle riscrivere. In particolare consente di assegnare lo stesso stile a più componenti all'interno del documento HTML. L'inserimento del codice CSS nelle pagine web può essere effettuato in tre modi diversi:

- inserimento nel tag `<head>` di un collegamento ad un foglio di stile esterno, come nell'esempio seguente:

```
<head>  
  <link rel="stylesheet" type="text/css"  
    href="stylesheet.css" />
```

`</head>`

- inserimento nel tag `<head>` delle dichiarazioni css tra i tag `<style>` e `</style>`, come nell'esempio seguente:

```
<head>
  <style type="text/css">
    /*codice css*/
  </style>
</head>
```

- inserimento del codice CSS all'interno degli elementi, come nell'esempio seguente:

```
<tag style="css">/tag>
```

Generalmente un foglio di stile CSS è strutturato come una sequenza di regole, ovvero coppie costituite da un selettore e un blocco di dichiarazioni. La struttura standard è la seguente:

```
selettore {
  p1: v1,
  p2: v2
}
```

Il selettore può essere di diverso tipo. Ad esempio il selettore può fare riferimento a un tipo o tag (come ad esempio *body*) o a una classe o a un identificatore. Il blocco di dichiarazioni invece è costituito da una proprietà e un valore assegnato a quella proprietà. Alcuni esempi di proprietà sono *width* e *height*, che vengono utilizzate per impostare altezza e larghezza di un determinato elemento, *color*, che viene utilizzata per definire il colore del testo di un elemento, e *text-align*, che viene utilizzato per definire l'allineamento degli elementi.

## TypeScript

*TypeScript* è un linguaggio di programmazione fortemente tipizzato open source sviluppato da Microsoft. A differenza di JavaScript, TypeScript presenta tipi, classi e interfacce. Il linguaggio estende la sintassi di JavaScript in modo che qualunque programma scritto in quel linguaggio possa funzionare con TypeScript senza alcuna modifica. TypeScript presenta quindi un sistema di annotazione dei tipi, che consente il controllo di questi tipi durante la fase di

compilazione, anche se è comunque possibile utilizzare la tipizzazione dinamica di JavaScript. I tipi primitivi utilizzabili in Typescript sono *number*, *string* e *boolean*.

## C#

C# è un linguaggio di programmazione orientato a oggetti e indipendente dai tipi. La sintassi di C# prende spunto da vari linguaggi nati precedentemente, come C++ o Java. C# solitamente viene utilizzato per sviluppare molti tipi di applicazioni eseguite in .NET, la piattaforma di sviluppo ideata e sviluppata da Microsoft. La sintassi di base è molto simile a quella dei linguaggi C, C++ e Java. C#, per la web application *Rapporti di Lavoro*, viene utilizzato per lo sviluppo della parte server, con i Controller che verranno descritti nel prossimo capitolo.

### 2.1.3 Database - Microsoft SQL Server

#### Database

Un *Database* è un insieme di dati strutturati memorizzati elettronicamente in un sistema informatico. I database solitamente vengono controllati e amministrati da un sistema DBMS (*Database Management Sysyem*). Esistono diversi tipi di database, che vengono descritti di seguito.

**Database Relazionali** I database relazionali sono stati introdotti negli anni '80. All'interno di un database relazionale i dati sono organizzati sotto forma di insiemi di tabelle composte da colonne e record. Questo tipo di tecnologia offre la soluzione più efficiente e flessibile per accedere alle informazioni. Le tabelle del database relazionale spesso includono un identificatore univoco per ogni riga all'interno della tabella, chiamato chiave primaria. Questa chiave primaria consente di indicizzare i dati e può anche essere utilizzata per condividere valori tra più tabelle all'interno di un database. I dati che fanno riferimento ad altri e che vengono utilizzati per mettere in relazione tra di loro le tabelle sono chiamati chiavi esterne. Questa tipologia di database è quella utilizzata per la web application *Rapporti di Lavoro*. Il linguaggio utilizzato per la gestione di questa tipologia di database è *SQL* e verrà approfondito nei prossimi paragrafi.

**Database orientati agli oggetti** All'interno di un database orientato agli oggetti, le informazioni vengono rappresentate sotto forma di oggetti, come nel tipo di programmazione omonima. In un database orientato agli oggetti un set

di dati viene associato con tutti i suoi attributi ad un unico oggetto. Oltre agli attributi, negli oggetti vengono memorizzati anche i metodi, infatti, come nel metodo di programmazione, ogni oggetto ha determinate attività che può svolgere. A ogni oggetto viene assegnato automaticamente un'identificazione univoca, in modo da indirizzare e richiamare facilmente gli oggetti salvati.

**Database distribuiti** Un database distribuito è composto da almeno due file presenti in due sedi diverse. Il database può quindi essere memorizzato su più computer all'interno della stessa rete o anche in reti diverse.

**Data warehouse** Un data warehouse è un tipo di database progettato espressamente per query e analisi veloci e consiste in un repository centralizzato per i dati. In particolare questo tipo di database viene utilizzato per il supporto di attività di business intelligence, in particolare gli analytics.

**Database NoSQL** Un database NoSQL, detto anche non relazionale, consente di archiviare e manipolare dati non strutturati e semi-strutturati, a differenza di quanto accade con un database relazionale che definisce come devono essere composti tutti i dati inseriti nel database. Il punto di forza di questo tipo di database è quindi la possibilità di avere modelli di dati flessibili e a differenza dei database relazionali i dati correlati tra loro non devono essere suddivisi tra tabelle, ma i dati correlati sono annidati all'interno di un'unica struttura dati.

**Database grafici** Un database grafico memorizza i dati in termini di entità e relazioni tra le entità. Questo tipo di database quindi è mappato su grafici, tramite i quali le informazioni e le relative relazioni sono visualizzate chiaramente e memorizzate come un insieme di dati ampio e coerente.

**Database OLTP** OLTP è un database di analisi dei dati veloce progettato per far fronte a un gran numero di transazioni eseguite da più utenti.

## SQL

*SQL (Structured Query Language)* è un linguaggio di programmazione utilizzato dalla maggior parte dei database relazionali per l'esecuzione di query e la manipolazione e la definizione dei dati. SQL è stato progettato per eseguire le seguenti operazioni:

- *DDL (Data Definition Language)*, ovvero creare e modificare gli schemi del database;

- *DML (Data Manipulation Language)*, ovvero inserire, modificare e gestire dati memorizzati;
- *DQL (Data Query Language)*, ovvero interrogare i dati memorizzati;
- *DCL (Data Control Language)*, ovvero creare e gestire strumenti di controllo e accesso ai dati.

**Data Definition Language** *DDL* viene utilizzato per la creazione, la modifica o l'eliminazione degli oggetti in un database. Sono quindi i comandi che definiscono la struttura del database. Per eseguire questi comandi, l'utente che li esegue deve avere i permessi necessari, che vengono assegnati tramite il *DCL*. Il linguaggio è quindi utilizzato in fase di progettazione del database. Questi comandi possono agire su *Domini* (per creare altri tipi da assegnare agli attributi), *Schemi* (per creare una collezione di tutti gli oggetti che faranno parte della base di dati), *Database*, *Tabelle* e *Indici*. Di seguito vengono mostrati alcuni dei comandi del *DDL*.

- *CREATE DOMAIN/SCHEMA/DATABASE/TABLE/INDEX*, utilizzato per la creazione di nuovi domini, schemi, database, tabelle e indici;
- *DROP DOMAIN/SCHEMA/DATABASE/TABLE/INDEX*, utilizzato per la rimozione di domini, schemi, database, tabelle e indici esistenti;
- *ALTER DATABASE/TABLE*, utilizzato per la modifica di database e tabelle esistenti.

Nella creazione delle tabelle è possibile aggiungere anche dei vincoli, descritti di seguito:

- *NOT NULL*, vincolo che impone che l'attributo al quale è assegnato non può assumere valori nulli;
- *UNIQUE*, vincolo che impone che per l'attributo al quale è assegnato non possa comparire lo stesso valore su righe diverse;
- *PRIMARY KEY*, vincolo che indica qual è la chiave primaria della tabella, ovvero qual è l'attributo che identifica in modo univoco una riga all'interno di una tabella;
- *FOREIGN KEY*, vincolo che impone che uno o più attributi fanno riferimento a una chiave primaria di un'altra tabella. In questo modo, per l'attributo che viene chiamato chiave esterna (appunto, foreign key) è possibile inserire solo valori presenti nella tabella referenziata.



**Data Manipulation Language** *DML* viene utilizzato per l'inserimento, la modifica e l'eliminazione dei dati all'interno delle tabelle di un database. La struttura di questi dati viene definita con l'utilizzo del DDL. Per eseguire i comandi del DML, l'utente che li esegue deve avere i permessi necessari, assegnati tramite il *DCL*. Di seguito vengono mostrati alcuni dei comandi del *DML*:

- *INSERT*, che consente di inserire dati nelle tabelle. Le colonne di destinazione dei valori possono essere dichiarate oppure no nel comando. La sintassi del comando Insert è la seguente:

```
INSERT INTO nomeTabella (elencoColonne)
VALUES (elencoValori)
```

dove *elencoColonne* è l'elenco delle colonne in cui inserire i valori e *elencoValori* è l'elenco dei valori da inserire nella tabella, che devono essere inseriti rispettando l'ordine dei campi dichiarati nell'elenco colonne;

- *UPDATE*, che consente di modificare i dati presenti nelle tabelle. E' possibile effettuare degli update generici o con condizione. La sintassi del comando Update è la seguente:

```
UPDATE nomeTabella
SET nomeCampo = nuovoValore
WHERE condizione
```

dove dopo la parola chiave *SET* bisogna inserire l'assegnamento del nuovo valore al campo desiderato, mentre dopo la parola chiave *WHERE* vanno inserire le condizioni per effettuare una selezione sulle righe da modificare;

- *DELETE*, che consente di eliminare i dati nelle tabelle. Il comando può effettuare operazioni in modo generico cancellando tutte le righe della tabella oppure ponendo delle condizioni. La sintassi del comando Delete è la seguente:

```
DELETE FROM nomeTabella
WHERE condizione
```

dove dopo la parola chiave *WHERE* vanno inserire le condizioni per effettuare una selezione sulle righe da eliminare.

**Data Query Language** *DQL* viene utilizzato per la lettura e l'elaborazione dei dati presenti in un database. Questi dati vengono inseriti tramite il *DML*

con strutture che sono state create con il *DDL*. Per eseguire questi comandi, l'utente che li esegue deve avere i permessi necessari, che vengono assegnati tramite il *DCL*. Il comando principale per l'estrazione dei dati è *SELECT*. La sintassi è la seguente:

```
SELECT listaElementi  
FROM listaTabelle  
WHERE condizione  
[GROUP BY listaColonne]  
[ORDER BY listaColonne]
```

dove *listaElementi* è l'elenco dei campi da estrarre, *listaTabelle* è l'elenco delle tabelle da cui estrarre i dati, *condizione* è l'elenco delle condizioni che un campo deve rispettare per potere essere selezionato dalla query e *listaColonne* è l'elenco di colonne di riferimento per l'ordinamento dei dati. La clausola *GROUP BY* viene utilizzata per raggruppare i risultati mentre la clausola *ORDER BY* viene utilizzato per ordinarli.

**Data Control Language** *DCL* viene utilizzato per concedere o revocare agli utenti i permessi necessari per poter utilizzare i comandi *DML* e *DDL*. I comandi *DCL* sono *Grant* e *Revoke*. *Grant* concede uno o più permessi ad un determinato utente, mentre *Revoke* li revoca.

### Microsoft SQL Server

Come già anticipato in precedenza, i database vengono controllati e amministrati da un DBMS (*Database Management System*), ovvero un sistema software progettato per consentire la creazione, la manipolazione e l'interrogazione di database. In particolare, per la web application *Rapporti di Lavoro* viene utilizzato *Microsoft SQL Server* che è un *DBMS* relazionale prodotto da Microsoft. *Microsoft SQL Server* usa una variante del linguaggio SQL standard, ovvero il *Transact-SQL* o *T-SQL*. *T-SQL* è l'estensione proprietaria del linguaggio *SQL* sviluppata da Microsoft. In aggiunta all'*SQL* standard, *T-SQL* aggiunge funzioni per controllo di flusso, definizione di variabili locali e funzioni per la manipolazione di stringhe o date.

Per quanto riguarda il controllo di flusso *T-SQL* implementa alcune keyword come ad esempio *IF* e *ELSE*, che consentono l'esecuzione condizionale di blocchi di istruzioni, *BEGIN* e *END*, che delimitano un blocco di istruzioni, o *WHILE*, che implementa dei cicli.

Invece, per quanto riguarda la definizione di variabili locali, *T-SQL* implemen-

ta l'utilizzo di variabili locali, accessibili solo allo script che le utilizza, mentre invece non è possibile implementare variabili globali. Le parole chiave relative all'utilizzo di variabili locali sono *DECLARE*, utilizzata per dichiarare una variabile, e *SET*, utilizzata per attribuirle un valore. Di seguito viene mostrato un esempio di come utilizzare queste parole chiave:

```
DECLARE @nomeVariabile tipoVariabile
SET @nomeVariabile = valore
```

## 2.2 Strumenti utilizzati

### 2.2.1 Microsoft Entity Framework Core

*Entity Framework Core* è una versione open source e multiplatforma della tecnologia di accesso ai dati *Entity Framework*, che consente agli sviluppatori di lavorare con i dati sotto forma di oggetti, senza doversi preoccupare della struttura del database in cui sono memorizzati questi dati e hanno quindi la possibilità di lavorare a un livello più alto di astrazione. Quindi, *Entity Framework Core*, o *EF Core*, può essere utilizzato come mapper relazionale a oggetti che consente agli sviluppatori .NET di usare un database usando semplicemente degli oggetti e elimina la necessità della maggior parte del codice di accesso ai dati.

Con *EF Core*, l'accesso ai dati viene eseguito tramite un modello, costituito da classi di entità e da un oggetto contesto (*Context*) che rappresenta una sessione con il database. L'oggetto *Context* consente l'esecuzione di query e il salvataggio dei dati. Tramite questo oggetto è quindi possibile eseguire query ai dati memorizzati nel database e salvarne di nuovi.

Per quanto riguarda l'eseguire query, è possibile effettuare i vari comandi appartenenti al *DQL*. Di seguito viene mostrato un esempio per l'esecuzione di una query:

```
using (var context = new DbContext())
{
    var dati = context.Entity
        .Where( x => x.id == 1).ToList();
}
```

All'interno di questo esempio la variabile *context* è il riferimento al database e *Entity* è la classe di riferimento alla entità desiderata. Con il metodo *Where* è possibile applicare dei filtri ai dati e con il metodo *ToList* i dati che soddisfano

le condizioni inserite nel metodo *Where* vengono restituiti in un oggetto *List* che rappresenta un elenco di oggetti di un determinato tipo.

Invece, per quanto riguarda il salvataggio di dati, è possibile eseguire i vari comandi appartenenti al *DML*. Di seguito viene mostrato un esempio:

```
using (var context = new DbContext())
{
    var nuovoDato = new Entity();

    context.Entity.Add(nuovoDato);
    context.SaveChanges();
}
```

Questo esempio mostra come inserire un nuovo elemento all'interno della tabella *Entity*. Infatti, con il metodo *Add* è possibile inserire l'oggetto *nuovoDato* nella tabella *Entity*. Invece, il metodo *SaveChanges* applica le modifiche al database. Senza l'utilizzo di questo metodo gli aggiornamenti apportati al database vengono annullati.

## Capitolo 3

# Implementazione

### 3.1 Modelli

All'interno di questo paragrafo vengono illustrate le classi più significative utilizzate nella web application *Rapporti di Lavoro* e i relativi attributi.

#### 3.1.1 AnalisiChimiche

La classe *AnalisiChimiche* rappresenta il relativo oggetto nel database. Questo oggetto viene utilizzato per la rappresentazione delle rilevazioni delle Analisi Chimiche relative alle colate specifiche. Queste analisi chimiche vengono rilevate tramite uno strumento, chiamato quantometro, e memorizzate su dei file. Questi file vengono poi analizzati dalla web application *Rapporti di Lavoro* e i dati relativi alle analisi chimiche vengono memorizzati nel database. I dati presenti in questo oggetto sono i seguenti:

- *AcId*, ovvero l'identificativo progressivo dell'analisi chimica;
- *AcIdColata*, ovvero il riferimento all'identificativo progressivo della colata a cui fa riferimento l'analisi;
- *AcIdMateriaPrima*, ovvero il riferimento all'identificativo progressivo del materiale rilevato durante l'analisi;
- *AcPercRilevata*, ovvero la quantità, in percentuale, dell'elemento rilevato presente all'interno della colata;
- *AcData*, ovvero data e ora in cui è stata effettuata la rilevazione;

- *AcIdQuantometro*, ovvero l'identificativo del quantometro, cioè lo strumento utilizzato per eseguire l'analisi chimica;
- *AcPerCliente*, ovvero un flag che indica se l'analisi è l'analisi definita *Per Cliente*, ovvero l'analisi definitiva;
- *AcDataPerCliente*, ovvero data e ora in cui è stata salvata l'analisi *Per Cliente*.

### 3.1.2 Anomalie

La classe *Anomalie* rappresenta il relativo oggetto nel database. Questo oggetto viene utilizzato per la rappresentazione delle anomalie che si sono verificate durante una colata nei vari impianti. Nelle pagine relative ai rapporti di lavoro è presente una tabella che consente la visualizzazione di questi dati e la possibilità di aggiungere delle nuove anomalie tramite un popup che consente l'inserimento della data in cui l'anomalia si è verificata e il tipo e la descrizione dell'anomalia. I dati presenti in questo oggetto sono i seguenti:

- *AId*, ovvero l'identificativo progressivo dell'anomalia;
- *AOperatore*, ovvero l'operatore che ha salvato l'anomalia;
- *AData*, ovvero data e ora in cui si è verificata l'anomalia;
- *AIdTipoAnomalia*, ovvero il riferimento all'identificativo progressivo del tipo di anomalia;
- *ADescrizione*, ovvero eventuali note che l'operatore inserisce durante la fase di salvataggio dell'anomalia;
- *AIdImpianto*, ovvero il riferimento all'identificativo dell'impianto in cui si è verificata l'anomalia;
- *AIdColataBase*, ovvero il riferimento all'identificativo progressivo della colata base in corso nel momento in cui si è verificata l'anomalia;
- *AIdColataSpecifica*, ovvero il riferimento all'identificativo progressivo della colata specifica in corso nel momento in cui si è verificata l'anomalia;
- *AIdMacchina*, ovvero il riferimento all'identificativo progressivo della macchina nella quale si è verificata l'anomalia, utilizzato solo nei *Rapporti Lavoro Colata Continua*.

### 3.1.3 Colate

La classe *VColateAll* rappresenta il relativo oggetto nel database. Questo oggetto viene utilizzato per la rappresentazione di tutte le informazioni relative alle colate (base e specifiche) che sono state schedate. Queste informazioni vengono visualizzate nella pagina *Storico Colate*. I dati presenti in questo oggetto sono i seguenti:

- *VcaId*, ovvero l'identificativo progressivo della colata;
- *VcaIdColataLegaBase*, ovvero il riferimento all'identificativo progressivo della colata base, utilizzato solo per le colate specifiche;
- *VcaIdLega*, ovvero il riferimento all'identificativo della lega prodotta dalla colata;
- *VcaNomeLega*, ovvero la descrizione relativa alla lega prodotta dalla colata;
- *VcaNumeroColata*, ovvero il numero della colata;
- *VcaIdDestinazione*, ovvero il riferimento all'impianto di destinazione della colata, cioè il forno verso il quale la colata base viene sversata;
- *VcaNomeDestinazione*, ovvero il nome della destinazione della colata;
- *VcaInizioSched*, ovvero data e ora prevista di inizio della colata;
- *VcaFineSched*, ovvero data e ora prevista di fine della colata;
- *VcaInizioOper*, ovvero data e ora effettive di inizio della colata;
- *VcaFineOper*, ovvero data e ora effettive di fine colata;
- *VcaIdStato*, ovvero il riferimento all'identificativo dello stato della colata;
- *VcaStato*, ovvero la descrizione relativa allo stato della colata;
- *VcaOperatoreColataPronta*, ovvero l'operatore che ha dichiarato la colata pronta;
- *VcaProntaOper*, ovvero data e ora in cui la colata è stata dichiarata pronta;
- *VcaOperatoreFineColata*, ovvero l'operatore che ha dichiarato la colata conclusa;
- *VcaPeso*, ovvero il peso totale della colata;

- *VcaNumColataProg*, ovvero il numero progressivo che identifica la colata, utilizzato solo per le colate specifiche;
- *VcaIsBase*, ovvero un flag che indica se la colata è una colata base;
- *VcaIsSpecifica*, ovvero un flag che indica se la colata è una colata specifica;
- *VcaIdCliente*, ovvero il riferimento all'identificativo del cliente relativo alla colata;
- *VcaNomeCliente*, ovvero il nome del cliente relativo alla colata.

### 3.1.4 Correzioni

La classe *Correzioni* rappresenta il relativo oggetto nel database. Questo oggetto viene utilizzato per la rappresentazione delle correzioni da effettuare alla colata in base ai calcoli effettuati sulle analisi chimiche. Nella pagina *Analisi Chimiche* è infatti possibile effettuare i calcoli delle correzioni per le analisi chimiche che non rispettano la specifica. Quando queste proposte di correzioni vengono accettate, le rispettive informazioni vengono memorizzate tramite questo oggetto. I dati presenti in questo oggetto sono i seguenti:

- *CId*, ovvero l'identificativo progressivo della correzione;
- *CIdMateria*, ovvero il riferimento all'identificativo del correttivo che deve essere utilizzato per eseguire la correzione;
- *CQtaDaCaricare*, ovvero la quantità del correttivo da caricare nella colata;
- *COperatore*, ovvero il nome dell'operatore che ha accettato la correzione;
- *CData*, ovvero data e ora in cui è stata accettata la proposta di correzione;
- *CIdAnalisi*, ovvero il riferimento all'identificativo dell'analisi chimica di riferimento;
- *CStatoCarica*, ovvero il riferimento all'identificativo dello stato carica della correzione;
- *CIdColata*, ovvero il riferimento all'identificativo della colata della quale deve essere effettuata la correzione.



### 3.1.5 Schede Output

La classe *SchedeOutput* rappresenta il relativo oggetto nel database. Questo oggetto viene utilizzato per la rappresentazione delle schede contenenti le informazioni relative agli output delle varie colate. Ogni scheda equivale a un cassone presente in impianto che contiene tutti gli output relativi alle colate del relativo impianto. I dati presenti in questo oggetto sono i seguenti:

- *SoId*, ovvero l'identificativo progressivo della scheda;
- *SoStato*, ovvero il riferimento all'identificativo dello stato della scheda;
- *SoNumeroCamion*, ovvero il progressivo del camion utilizzato per la spedizione del contenuto del cassone collegato alla scheda;
- *SoAperturaScheda*, ovvero data e ora di apertura della scheda output;
- *SoSchedaPronta*, ovvero data e ora di chiusura della scheda output;
- *SoSchedaSpedita*, ovvero data e ora di spedizione della scheda output;
- *SoTipoScheda*, ovvero il riferimento all'identificativo del tipo di scheda;
- *SoOperatoreAperturaScheda*, ovvero l'operatore che ha aperto la scheda;
- *SoOperatoreSchedaPronta*, ovvero l'operatore che ha dichiarato chiusa la scheda;
- *SoOperatoreSchedaSpedita*, ovvero l'operatore che ha dichiarato la spedizione della scheda;
- *SoCassone*, ovvero il riferimento all'identificativo progressivo del cassone di riferimento;
- *SoNumeroDdt*, ovvero il numero del documento di trasporto (*DDT*) relativo alla spedizione della scheda.

### 3.1.6 Output Colata

La classe *VOutputColata* rappresenta il relativo oggetto nel database. Questo oggetto viene utilizzato per la rappresentazione delle informazioni relative agli output di tutte le colate, base e specifiche. I dati presenti in questo oggetto sono i seguenti:

- *VocId*, ovvero l'identificativo progressivo dell'output della colata;

- *VocIdScheda*, ovvero il riferimento all'identificativo della scheda in cui è memorizzato l'output;
- *VocQuantitaOutput*, ovvero la quantità totale dell'output;
- *VocTipoOutput*, ovvero l'identificativo del tipo di output;
- *VocDescOutput*, ovvero il descrittivo del tipo di output;
- *VocDataColata*, ovvero data e ora di inizio della colata;
- *VocDichiarazione*, ovvero data e ora in cui è stato inserito l'output;
- *VocNumColataProg*, ovvero il numero progressivo della colata di riferimento;
- *VocNumColata*, ovvero il numero della colata di riferimento;
- *VocIdColata*, ovvero il riferimento all'id della colata relativa;
- *VocSettimana*, ovvero il numero della settimana nell'anno in cui è stata effettuata la scorifica;
- *VocNomeLega*, ovvero il nome della lega relativa alla colata di riferimento;
- *VocScorifica*, ovvero data e ora in cui è stata effettuata la scorifica;
- *VocOperatore*, ovvero l'operatore che ha inserito l'output;
- *VocIdQualita*, ovvero il riferimento all'id della qualità dell'output;
- *VocNomeQualita*, ovvero il descrittivo della qualità dell'output;
- *VocIdProvenienzaColaticcio*, ovvero il riferimento all'id della provenienza dell'output;
- *VocNomeProvenienzaColaticcio*, ovvero il descrittivo della provenienza dell'output;
- *VocNote*, ovvero eventuali note inserite dall'operatore che ha aggiunto l'output.

### 3.1.7 Rapporti Lavoro

La classe *VRapportiLavoro* rappresenta il relativo oggetto nel database. Questo oggetto viene utilizzato per la rappresentazione delle informazioni relative ai rapporti di lavoro visualizzate nelle rispettive pagine. Questo oggetto rappresenta sia i rapporti di lavoro relativi alle colate base utilizzati nei rapporti di lavoro del forno fusorio, sia i rapporti di lavoro relativi alle colate specifiche utilizzati nei rapporti di lavoro dei forni bacino, della colata continua e del magazzino pani. I dati presenti in questo oggetto sono i seguenti:

- *VrId*, ovvero l'identificativo progressivo della colata di riferimento;
- *VrDataInizio*, ovvero data e ora di inizio della colata di riferimento;
- *VrDataFine*, ovvero data e ora di fine della colata di riferimento;
- *VrIdDestinazione*, ovvero il riferimento all'identificativo della destinazione della colata;
- *VrDestinazioneNome*, ovvero il descrittivo della destinazione di riferimento;
- *VrOperatore*, ovvero l'operatore che ha dato inizio alla colata;
- *VrNumeroColata*, ovvero il numero della colata di riferimento;
- *VrAnno*, ovvero l'anno in cui la colata è stata prodotta;
- *VrLega*, ovvero la lega prodotta dalla colata di riferimento;
- *VrDurata*, ovvero la durata della colata di riferimento;
- *VrBase*, ovvero un flag che indica se il rapporto fa riferimento a una colata base o specifica;

## 3.2 Controller

All'interno di questo paragrafo vengono illustrati i controller utilizzati nella web application *Rapporti di Lavoro* e i relativi metodi più importanti.

### 3.2.1 AnalisiChimicheController

La classe *AnalisiChimicheController* contiene tutte le funzioni relative alle Analisi Chimiche, come le funzioni per il calcolo delle correzioni o per il salvataggio delle proposte di correzioni. I metodi più significativi di questa classe sono i seguenti:

- *AnalisiFile*. Questo metodo viene eseguito ogni minuto e analizza i file csv memorizzati in una determinata cartella (il quale percorso è configurabile). Questi file vengono analizzati per memorizzare nel database i dettagli dell'analisi chimica che ha generato il file stesso. I dati letti nel file e salvati nel database sono l'id del quantometro, il numero della colata, il tipo di analisi, l'id dell'elemento chimico e la percentuale rilevata. Inoltre, se l'elemento chimico non è presente nel relativo dizionario, questo viene aggiunto in automatico con dati impostati di default, poi modificabili dalla sezione *Materie e Elementi*;
- *GetChemicalForColata*. Questo metodo prende come parametro l'id e la destinazione della colata della quale ottenere la lista delle relative analisi chimiche da visualizzare nella pagina *Analisi Chimiche*. Il metodo restituisce un array contenente tutti i dati letti dal database, di tutte le analisi chimiche relative alla colata passata come parametro;
- *GetElencoAnalisi*. Questo metodo prende come parametri l'id dell'impianto, l'id e il peso della colata e l'id dell'analisi desiderata e restituisce i relativi dettagli, come l'elemento, la percentuale rilevata, il peso (calcolato in base alla percentuale e al peso totale della colata) e le specifiche da rispettare per rimanere all'interno dello standard. Il metodo restituisce un oggetto contenente i dettagli dell'analisi e delle correzioni eventualmente già accettate e memorizzate nel database;
- *CalcoloPropostaCorrezioni*. Questo metodo prende come parametri l'elenco delle rilevazioni relative a un'analisi chimica e restituisce un oggetto contenente le proposte di correzione per quella determinata analisi chimica. In questo metodo vengono calcolate le quantità dei correttivi da utilizzare per adattare le quantità degli elementi chimici agli standard produttivi.

### 3.2.2 ColateController

La classe *ColateController* contiene tutte le funzioni relative alle colate, come le funzioni per la lettura delle colate attualmente in corso o l'inserimento di nuove informazioni relative alla colata. I metodi più significativi di questa classe sono i seguenti:

- *GetActualColataFornoFusorio*. Questo metodo restituisce un oggetto di tipo *VColateAll* contenente le informazioni relative alla colata attualmente in corso nel Forno Fusorio;
- *GetActualColataForniBacino*. Questo metodo restituisce un oggetto di tipo *VColateAll* contenente le informazioni relative alla colata attualmente in corso nel Forno a Bacino passato come parametro al metodo;
- *AddPesoColata*. Questo metodo consente di salvare nel database il peso totale della colata passata come parametro;
- *GetColateBaseByDate*. Questo metodo restituisce un array di oggetti contenenti le informazioni relative alle colate base con data di inizio o di schedulazione compresa tra le date di inizio e di fine passate come parametro;
- *GetColateSpecificheByBase*. Questo metodo restituisce un array di oggetti contenenti le informazioni relative alle colate specifiche che fanno riferimento alla colata base passata come parametro;
- *GetLegheBaseByColata*. Questo metodo restituisce le informazioni della lega base relativa alla colata base passata come parametro;
- *GetLegheSpecificheByColata*. Questo metodo restituisce le informazioni della lega specifica relativa alla colata specifica passata come parametro.

### 3.2.3 OuptutColataController

- *AddOutputBase*. Questo metodo inserisce nel database la registrazione dell'output di una colata base passato come parametro;
- *MoveOutputBase*. Questo metodo cambia il riferimento di una registrazione di un output base da una scheda output a un'altra scheda passata come parametro;
- *AddOutputSpecifica*. Questo metodo inserisce nel database la registrazione dell'output di una colata specifica passato come parametro;

- *MoveOutputSpecifica*. Questo metodo cambia il riferimento di una registrazione di un output specifico da una scheda output a un'altra scheda passata come parametro;
- *GetOutputColataBaseByScheda*. Questo metodo restituisce gli output base che fanno riferimento alla scheda passata come parametro;
- *GetOutputColataSpecificaByScheda*. Questo metodo restituisce gli output specifici che fanno riferimento alla scheda passata come parametro;

### 3.2.4 RapportiDiLavoroController

- *GetRapportoFornoFusorioActualColata*. Questo metodo restituisce il rapporto di lavoro base in corso attualmente;
- *GetRapportoForniBacinoActualColata*. Questo metodo restituisce il rapporto di lavoro relativa alla colata attualmente in corso nel forno a bacino passato come parametro;
- *GetRapportoColataContinuaActualColata*. Questo metodo restituisce il rapporto di lavoro relativa alla colata attualmente in corso nella colata continua;
- *GetRapportiColataContinuaByYear*. Questo metodo restituisce l'elenco di tutti i rapporti specifici della colata continua ordinati per data in modo decrescente e filtrati in base all'anno passato come parametro;
- *GetRapportiMagazzinoPaniByYear*. Questo metodo restituisce l'elenco di tutti i rapporti specifici del magazzino pani ordinati per data in modo decrescente e filtrati in base all'anno passato come parametro;
- *GetFermiFornoFusorio*. Questo metodo restituisce la lista di fermi che si sono verificati all'interno del turno passato come parametro;
- *GetFermiColataContinuaByColata*. Questo metodo restituisce la lista di fermi che si sono verificati durante la colata passata come parametro;
- *GetFermiMagazzinoPaniByColata*. Questo metodo restituisce la lista di fermi che si sono verificati durante la colata passata come parametro;
- *AddFermo*. Questo metodo consente di aggiungere un nuovo fermo manualmente, con data e ora di inizio e di fine e i cinque perchè passati come parametro;

- *EditListaFermo*. Questo metodo consente di modificare il fermo passato come parametro e viene utilizzato per causalizzare i fermi automatici salvati dal PLC;
- *AddAnomalia*. Questo metodo consente di inserire nel database una nuova anomalia che viene passata come parametro;
- *GetAnomalieFornoFusorioByShift*. Questo metodo restituisce l'elenco di anomalie che si sono verificate nel forno fusorio durante il turno passato come parametro;
- *GetAnomalieColataContinuaByColata*. Questo metodo restituisce l'elenco di anomalie che si sono verificate nel reparto di colata continua durante la colata passata come parametro;
- *GetAnomalieMagazzinoPaniByColata*. Questo metodo restituisce l'elenco di anomalie che si sono verificate nel reparto magazzino pani durante la colata passata come parametro;
- *GetAnomalieForniBacinoByColata*. Questo metodo restituisce l'elenco di anomalie che si sono verificate nel forno a bacino durante la colata e nel forno passati come parametro;
- *AddRallentamento*. Questo metodo consente di inserire nel database un nuovo rallentamento che viene passato come parametro;
- *GetRallentamentiColataContinua*. Questo metodo restituisce l'elenco di rallentamenti che si sono verificati nel reparto di colata continua durante la colata passata come parametro;
- *GetNoteFornoFusorioByShift*. Questo metodo restituisce l'elenco di note relative al forno fusorio che sono state inserite durante il turno passato come parametro;
- *GetNoteColataContinuaActualColata*. Questo metodo restituisce l'elenco di note relative alla colata continua che sono state inserite durante la colata passata come parametro;
- *GetNoteColataContinuaPrevTenColate*. Questo metodo restituisce l'elenco di note relative alla colata continua che sono state inserite durante le ultime dieci colate precedenti a quella passata come parametro;

- *GetNoteMagazzinoPaniActualColata*. Questo metodo restituisce l'elenco di note relative al magazzino pani che sono state inserite durante la colata passata come parametro;
- *GetNoteMagazzinoPaniPrevTenColate*. Questo metodo restituisce l'elenco di note relative al magazzino pani che sono state inserite durante le ultime dieci colate passate come parametro;
- *GetNoteForniBacinoByShift*. Questo metodo restituisce l'elenco di note relative al forno bacino che sono state inserite durante il turno passato e nel forno passati come parametro;
- *AddNota*. Questo metodo consente di inserire una nuova nota per il reparto passato come parametro;
- *GetPassaggioConsegneColataContinua*. Questo metodo restituisce l'elenco di passaggi di consegne relative alla colata continua che sono stati inseriti durante la colata passata come parametro;
- *GetPassaggioConsegneMagazzinoPani*. Questo metodo restituisce l'elenco di passaggi di consegne relative al magazzino pani che sono stati inseriti durante la colata passata come parametro;
- *GetPassaggioConsegneColataContinuaPrevShift*. Questo metodo restituisce l'elenco di passaggi di consegne relative alla colata continua che sono stati inseriti durante il turno precedente a quello attuale, passato come parametro;
- *GetPassaggioConsegneMagazzinoPaniPrevShift*. Questo metodo restituisce l'elenco di passaggi di consegne relative al magazzino pani che sono stati inseriti durante il turno precedente a quello attuale, passato come parametro;
- *GetPaniDaRifondereListByColata*. Questo metodo restituisce l'elenco delle rilevazioni dei pani da rifondere che sono state inserite durante la colata passata come parametro;
- *AddPaniDaRifondere*. Questo metodo consente di inserire una nuova rilevazione di pani da rifondere che viene passata come parametra;
- *GetInterruzioniMagazzinoPani*. Questo metodo restituisce l'elenco di interruzioni verificatesi nel magazzino pani relative alla colata passata come parametro;



- *AddInterruzione*. Questo metodo consente di inserire una nuova interruzione all'orario passato come parametro;
- *CloseInterruzione*. Questo metodo consente di chiudere l'interruzione in corso impostando come orario di fine quello passato come parametro;
- *GetNumeroFormatoPacchi*. Questo metodo restituisce il numero e il formato dei pacchi che vengono analizzati nel reparto magazzino pani. Il metodo viene richiamato ogni minuto e interroga un servizio esterno chiamato WorkerMsgScadaService, sempre sviluppato dal team di *Adipso*, per ricevere le informazioni direttamente dal server SCADA;
- *GetTempiComponenti*. Questo metodo restituisce le informazioni sul tempo di lavoro dei componenti nel reparto di colata continua. Il metodo viene richiamato ogni dieci minuti e interroga il servizio esterno WorkerMsgScadaService per ricevere le informazioni direttamente dal server SCADA;
- *ResetOreLavoro*. Questo metodo consente di resettare le informazioni sul tempo di lavoro dei componenti nel reparto di colata continua. Il metodo invia al servizio esterno WorkerMsgScadaService la richiesta di reset dei tag e restituisce l'esito dell'operazione.

### 3.2.5 SchedeOutputController

- *AddNewScheda*. Questo metodo consente di creare una nuova scheda associata a un determinato cassone. Questo metodo viene richiamato in automatico ogni volta che una scheda viene chiusa, per creare una nuova scheda relativa allo stesso cassone di quella precedente. Inoltre questo metodo viene richiamato anche su richiesta dell'utente per inserire una nuova scheda solo per un cassone che non ha alcuna scheda associata;
- *CloseScheda*. Questo metodo consente di chiudere la scheda passata come parametro e di dichiararla *IN ATTESA DI SPEDIZIONE*;
- *AddNCamion*. Questo metodo consente di dichiarare una scheda come spedita e a inserire il numero progressivo del camion che ha spedito la scheda;
- *AddNDdt*. Questo metodo consente di inserire le informazioni relative al documento di trasporto associato alla spedizione di una determinata scheda output passata come parametro;

- *GetSchedaAttivaByCassone*. Questo metodo restituisce le informazioni relative alla scheda attualmente aperta relativa la cassone passato come parametro;
- *GetSchedeBaseChiuseByDate*. Questo metodo restituisce l'elenco di tutte le schede base chiuse comprese nel periodo di tempo passate come parametro;
- *GetSchedeBaseAperte*. Questo metodo restituisce le informazioni relative a tutte le schede base aperte;
- *GetSchedeSpecificheChiuseByDate*. Questo metodo restituisce l'elenco di tutte le schede specifiche chiuse comprese nel periodo di tempo passate come parametro;
- *GetSchedeSpecificheAperte*. Questo metodo restituisce le informazioni relative a tutte le schede specifiche aperte;
- *GetSchedaById*. Questo metodo restituisce le informazioni relative alla scheda che ha come identificativo il valore passato come parametro.

### 3.2.6 ShiftController

- *GetActualShift*. Questo metodo restituisce le informazioni relative al turno in corso in base all'utente passato come parametro e al tipo di turno dello stesso utente. Le informazioni restituite sono la data di inizio, di fine e il numero del turno nella giornata;
- *GetPreviousShift*. Questo metodo restituisce le informazioni relative al turno precedente a quello attuale base all'utente passato come parametro e al tipo di turno dello stesso utente. Le informazioni restituite sono la data di inizio, di fine e il numero del turno nella giornata;
- *GetOperatoriASupporto*. Questo metodo restituisce l'elenco degli operatori a supporto dell'operatore principale nell'impianto e nel turno passati come parametro.

### 3.2.7 UserController

- *AuthenticateSupportUser*. Questo metodo consente di effettuare il login degli operatori di supporto, ovvero operatori che assistono gli operatori primari nei diversi reparti;

- *AuthenticateUser*. Questo metodo consente di effettuare il login degli operatori che viene effettuato all'apertura della web application. Il login viene effettuato in base alla tipologia di security impostata nel file di configurazione. Di default è impostata la tipologia *Domain*, quindi le credenziali da inserire durante la fase di login sono legate al dominio a cui appartiene il server sul quale è installata la web application;
- *LoadAllFunctByUserGroup*. Questo il metodo viene utilizzato per recuperare dal database le funzionalità a cui ha accesso un determinato utente in base al suo gruppo di appartenenza, che viene passato come parametro.

## Capitolo 4

# Interfaccia

4.1 Login

4.2 Storico Colate

4.3 Schede Colaticci

4.4 Rapporti Di Lavoro Forno Fusorio

4.5 Rapporti Di Lavoro Forni a Bacino

4.6 Rapporti Di Lavoro Colata Continua

4.7 Rapporti Di Lavoro Magazzino Pani

4.8 Analisi Chimiche

4.9 Materie e Elementi Chimici

4.10 Gestione Dizionari

4.11 Security

## Capitolo 5

# Sviluppi futuri e conclusione

### 5.1 Sviluppi futuri

#### 5.1.1 Analisi Metallografiche

#### 5.1.2 Gestione carichi magazzino

### 5.2 Conclusioni