

Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA "TULLIO LEVI-CIVITA "

CORSO DI LAUREA IN INFORMATICA



**Sviluppo di algoritmi euristici di ottimizzazione e
applicazione alla pianificazione della produzione**

Tesi di laurea triennale

Relatore

Prof. Luigi De Giovanni

Laureando

Riccardo Basso

ANNO ACCADEMICO 2018-2019

Sommario

Il presente documento descrive il lavoro svolto durante il periodo di stage, della durata di circa trecento ore, dal laureando Pinco Pallino presso l'azienda Azienda S.p.A. Gli obbiettivi da raggiungere erano molteplici.

In primo luogo era richiesto lo sviluppo di ... In secondo luogo era richiesta l'implementazione di un ... Tale framework permette di registrare gli eventi di un controllore programmabile, quali segnali applicati Terzo ed ultimo obbiettivo era l'integrazione ...

“Life is really simple, but we insist on making it complicated”

— Confucius

Ringraziamenti

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

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

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

Padova, Dicembre 2019

Riccardo Basso

Indice

1	Introduzione	1
1.1	L'azienda	1
1.1.1	Organizzazione aziendale	1
1.1.2	Servizi aziendali	1
1.1.3	Ergdis	3
1.1.4	Azienda e stage	4
1.1.5	Progetto scelto: Pianificazione della produzione	5
1.1.6	Convenzioni tipografiche	6
2	Descrizione dello stage	7
2.1	Panoramica del progetto	7
2.2	Specifiche tecniche del problema	7
2.2.1	Implementazione	8
2.2.2	Problemi logici da affrontare	9
2.3	Obiettivi	10
2.3.1	Requisiti obbligatori	11
2.3.2	Requisiti desiderabili	11
2.3.3	Requisiti opzionali	11
2.4	Divisione settimanale	11
2.5	Analisi dei rischi	14
3	Progetto di stage	15
3.1	Analisi dei requisiti	15
3.1.1	Casi d'uso	15
3.2	Tracciamento dei requisiti	17
3.2.1	Requisiti funzionali	18
3.2.2	Requisiti qualitativi	19
3.2.3	Requisiti prestazionali	20
3.2.4	Requisiti vincolo	20
3.2.5	Riepilogo dei requisiti	21
3.3	Tecnologie e strumenti	22
4	Progettazione e sviluppo	25
4.1	Progettazione	25
4.1.1	Architettura	25
4.1.2	Algoritmo	26
5	Progettazione e codifica	29
5.1	Tecnologie e strumenti	29

5.2	Ciclo di vita del software	29
5.3	Progettazione	29
5.4	Design Pattern utilizzati	29
5.5	Codifica	29
6	Verifica e validazione	31
7	Conclusioni	33
7.1	Consuntivo finale	33
7.2	Raggiungimento degli obiettivi	33
7.3	Conoscenze acquisite	33
7.4	Valutazione personale	33
A	Appendice A	35
	Bibliografia	39

Elenco delle figure

2.1	Requisiti	10
3.1	Use Case - UC-3	16
3.2	Use Case - UC-3.3	17
3.3	Logo Vb.NET	22
3.4	Logo Visuali Studio 2010	23
3.5	Logo DevExpress	23
3.6	Logo JSON	24
3.7	Logo Informix	24
4.1	Architettura generale	25

Elenco delle tabelle

2.5.1	Tabella dei rischi di progetto	14
3.2.1.	Tabella dei requisiti funzionali	18
3.2.2.	Tabella dei requisiti qualitativi	19
3.2.3.	Tabella dei requisiti prestazionali	20
3.2.4.	Tabella dei requisiti prestazionali	20
3.2.5.	Tabella del riepilogo dei requisiti	21

Capitolo 1

Introduzione

1.1 L'azienda

Ergon Informatica viene fondata nel 1988 come società di Ingegneria Informatica per applicazioni gestionali dedicate.

La società, che all'inizio conta solo alcuni dipendenti, si sviluppa in maniera costante negli anni e oggi può vantare una posizione di tutto rispetto tra le aziende dello stesso settore.

I clienti iniziali hanno giocato un ruolo fondamentale nello sviluppo del software prodotto da Ergon; essi infatti appartenevano per la maggior parte all'ambito alimentare e l'esperienza di queste prime installazioni ha permesso di acquisire delle competenze interne altamente specializzate in questo settore.

1.1.1 Organizzazione aziendale

Attualmente fanno parte della stessa gestione tre società:

- Ergon Informatica S.r.l. che si occupa dello sviluppo software, in particolare del gestionale Ergdis;
- Ergon S.r.l. che si occupa dei servizi tecnologici;
- Ergon Servizi S.r.l. che si occupa dei servizi amministrativi, logistici e di marketing delle sopracitate.

1.1.2 Servizi aziendali

- **Help Desk:** Il servizio di assistenza è fornito tramite un'apertura di chiamata che può essere effettuata via web o telefonando presso la sede. Un servizio di monitoring interno permette di individuare gli eventuali ritardi nelle risposte e negli interventi risolutivi. Il sistema di qualità (ISO 9001), che Ergon ha adottato, individua un tempo massimo di risoluzione del problema. Il cliente, collegandosi all'area a lui riservata, ha la possibilità di monitorare tutte le sue chiamate e verificare, per quelle ancora aperte, il tempo previsto di chiusura;
- **Servizio di Monitoring:** Il servizio prevede l'installazione per ogni server di un certo numero di agenti. Un operatore di Ergon, da remoto, attraverso un programma di controllo, rileva i vari tipi di "alert" e, in accordo con il cliente, può

intervenire se è necessario eliminare il problema. Alcuni esempi di alert possono essere: la memoria quasi esaurita, i salvataggi non effettuati;

- **Vendita e installazione Hardware:** Il servizio prevede la vendita, installazione e configurazione di qualsiasi tipo di apparato informatico direttamente presso la sede del cliente;
- **Radio Frequenza:** Il servizio prevede il supporto all'utilizzo di apparati radio per la comunicazione;
- **Sicurezza:** Il servizio prevede la vendita, installazione e aggiornamento costante dei servizi di sicurezza, affiliandosi a dei vendor internazionali;
- **Virtualizzazione:** Il servizio prevede la virtualizzazione dei server in modo da ottimizzare la performance dell'infrastruttura attraverso la creazione di server virtuali che sostituiscono quelli fisici. Ergon Informatica è specializzata nell'installazione del software di VMWARE e nel corso degli anni ha virtualizzato l'infrastruttura di quasi tutti i suoi clienti. Per il backup dei dati, in ambiente virtualizzato, utilizziamo il software di VEEAM;
- **Networking:** Il servizio prevede la progettazione, installazione e mantenimento di reti informatiche delle aziende;
- **Servizi Cloud:** Il servizio SLA (Service Level Agreement), servizio cloud offerto da Ergon Informatica viene erogato al cliente tramite un collegamento ai server che risiedono in un datacenter con elevati livelli di sicurezza ,può essere acquistato pagando un canone mensile per posto di lavoro; il cliente può quindi valutare in modo semplice i suoi costi e, grazie alla scalabilità del sistema, adattare in qualsiasi momento la struttura alle sue esigenze.

1.1.3 Ergdis

Ergdis è il sistema ERP progettato e sviluppato da Ergon Informatica S.r.l. L'insieme dei moduli proposti copre ogni aspetto della gestione aziendale:

- **Amministrazione e finanza:** È il prodotto che semplifica l'amministrazione aziendale, gestendo la contabilità e gli obblighi fiscali dell'azienda, offrendo analisi dettagliate sulla condizione creditizia e debitoria e fornendo un quadro in tempo reale della situazione contabile aziendale;
- **Controllo di Gestione:** È l'area applicativa del software Ergdis che supporta tutto il management nelle decisioni da prendere, promuovendo un'efficace supervisione della gestione interna dell'azienda. Permette un'analisi completa dei flussi economici che interessano la realtà aziendale, organizza i budget e i consuntivi, consente l'analisi dei costi di prodotto e rende funzionale l'organizzazione interna dell'impresa;
- **Area Acquisti:** È il prodotto che permette di sopperire ai bisogni di approvvigionamento della merce e dei servizi, attraverso un'efficace gestione degli ordini fornitori. Consente di ottimizzare l'intero processo relativo all'acquisto grazie ad un'amministrazione capillare delle disponibilità dei prodotti e dei documenti ad esso collegati. Infine garantisce il controllo dell'intero ciclo passivo: dall'ordine al fornitore fino alla ricezione della fattura;
- **Radio Frequenza:** Il servizio prevede il supporto all'utilizzo di apparati radio per la comunicazione;
- **Logistica:** È il prodotto che coordina l'insieme delle attività organizzative, gestionali e strategiche che regolano le movimentazioni di magazzino, consentendo la rintracciabilità della merce, monitorando le scorte dei prodotti, gestendo i documenti di carico e scarico. Grazie a questa suite di Ergdis l'utente potrà ottimizzare i propri costi e amministrare in maniera efficiente le operazioni che interessano l'entrata, l'uscita e la giacenza degli articoli;
- **Vendite:** È l'applicativo del software Ergdis realizzato per gestire l'area commerciale dell'azienda: dall'offerta iniziale fino alla fatturazione del prodotto, dagli ordini cliente alle campagne acquisti. Si compone di moduli software personalizzabili e modellabili secondo le esigenze, che semplificano i processi del business e migliorano l'utilizzo delle risorse aziendali. Grazie alla suite di programmi del Ciclo attivo sarà possibile velocizzare il passaggio dei dati e rendere automatici molti procedimenti prima manuali;
- **Produzione:** Questa suite di programmi offre un solido supporto nella pianificazione e nel controllo della produzione, ideale per chi necessita di analizzare con flessibilità le informazioni aziendali. Il prodotto consente infatti di gestire l'intero ciclo produttivo: dalla fase di programmazione, agli avanzamenti di produzione; dal versamento del prodotto finito al calcolo dei costi;
- **Web:** La suite di programmi consente di organizzare i propri negozi virtuali dando visibilità ai prodotti e gestendo le informazioni a questi legate, per un completo controllo delle esigenze aziendali;
- **Business Intelligence:** È il prodotto che consente di gestire graficamente le informazioni aziendali, effettuando analisi su tutti i dati gestionali, visualizzandoli

su griglie, dashboard e tabelle Pivot, in modo da renderli immediati e facilmente fruibili;

- **Qualità:** È il software che permette di garantire la qualità dell'azienda, attraverso un efficace sistema di controllo della soddisfazione dei clienti e della conformità dei prodotti venduti. Consente di realizzare le schede tecniche degli articoli, di creare dei questionari e di redigere i verbali delle riunioni, quest'ultime attività necessarie per chi possiede una certificazione di qualità. Garantisce infine l'assistenza ai clienti, grazie ad un software che guida l'utente nella risposta delle richieste e nella tempestiva risoluzione dei problemi;
- **Archiviazione Documentale:** L'utente può organizzare al meglio i propri documenti, rintracciandoli con facilità e snellendo il processo di ricerca informazioni. I moduli di quest'area permettono inoltre una migliore archiviazione del materiale in linea con le norme civili vigenti;
- **Gestione Archivi:** È il prodotto che favorisce un'ottima gestione dell'azienda attraverso la creazione di file e schede tecniche che codificano i prodotti, i clienti, le consegne e li organizzano secondo una struttura logica;
- **Pianificazione Consegne:** Permette di programmare e gestire le consegne, assegnandole ad un gruppo di mezzi, ottimizzando i percorsi compiuti dagli stessi nel rispetto dei vincoli definiti dal cliente.;
- **Area Mobile:** Proiettata anche verso il mondo mobile, Ergon ha ideato alcune app che lavorano in ambiente Android e permettono di gestire numerose attività dell'area commerciale dell'azienda;

1.1.4 Azienda e stage

Ergon Informatica propone già da diversi anni nuovi progetti di stage per gli studenti dell'Università degli Studi di Padova nella giornata di StageIT. Lo scopo di questa collaborazione è quello di creare un punto di incontro professionale con lo studente che si avvicina al mondo del lavoro. Entrambe le parti ne traggono beneficio: lo studente si cimenta in progetti aziendali che arricchiscono le sue conoscenze e ampliano le competenze acquisite in ambito scolastico, l'azienda invece valuta il possibile inserimento nel team di sviluppo dello studente coinvolto.

Progetti StageIT

- **App Entrata Merce terminali wifi:** Il progetto si propone di effettuare un'analisi approfondita per lo sviluppo di un'applicazione per l'entrata merce in magazzino. Il candidato dovrà valutare se sia più opportuno riscrivere l'applicazione su WEB o client/server in VB .Net, effettuando un'analisi approfondita dei vantaggi e degli svantaggi delle due scelte. Nel caso di applicazione Web, essa dovrà essere creata con appoggio su server WEB come fosse un sito, nel caso di applicazione VB .Net dovrà essere usata tramite l'appoggio di un server Windows terminal server. Dopo aver scelto la strada più opportuna, il candidato procederà con lo sviluppo dell'applicazione. Questa dovrà interfacciarsi con database Informix ed effettuare le normali operazioni necessarie alla movimentazione del magazzino;
- **App Customer Service:** Realizzazione di una app per il customer service, che permetta agli utenti di effettuare le richieste di assistenza o di anomalia anche

dallo smartphone. L'attuale software gestisce i tickets via web, con apertura di chiamata direttamente dalla propria area riservata. Attraverso questa app si vuole dare la possibilità all'utente di visualizzare tutte le richieste di assistenza, così come di crearne di nuove, anche in mobilità;

- **App Gestione Note spese:** Progettazione e sviluppo di una app mobile che permetta di gestire le informazioni relative alle note spese dei collaboratori. L'applicazione sincronizzerà i dati presenti sul dispositivo con l'archivio di gestione dell'ERP proprietario;
- **Project Manager:** Il Project Manager di Ergon vuole essere uno strumento per la pianificazione delle risorse e la distribuzione del carico lavoro in azienda, con lo scopo di organizzare al meglio task e compiti dei singoli. Una migliore pianificazione contribuirà ad una riduzione degli sprechi dovuti ad errate valutazioni e ad una maggiore puntualità nell'organizzazione interna. Allo stesso modo risorse meglio impiegate permetteranno all'azienda di essere tempestiva nei confronti dei clienti, rispondendo con più precisione alle loro richieste. Su più ampia scala il software sarà di aiuto per comprendere un eventuale fabbisogno di personale e sarà un valido strumento di gestione per il responsabile tecnico;
- **Pianificazione della produzione:** Il candidato dovrà integrare e potenziare l'algoritmo esistente, che consente di pianificare la produzione di un periodo temporale, razionalizzando l'occupazione delle linee e rispettando i tempi di spedizione degli ordini clienti. Il risultato dell'elaborazione indicherà gli articoli da produrre per soddisfare gli ordini clienti, programmando in quale giorno e fascia oraria realizzare le quantità necessarie. Il software dovrà inoltre, in presenza di eventi accidentali quali rottura della linea, guasto momentaneo, ordini dell'ultimo momento etc, trovare la soluzione ottimale ricalcolando l'impiego delle stesse.

1.1.5 Progetto scelto: Pianificazione della produzione

Dopo un'attenta valutazione delle varie proposte sopracitate, anche a seguito di varie discussioni con i responsabili dell'azienda, ho voluto scartare tutti i progetti che includessero lo sviluppo di un'applicazione mobile o servizio web. Questa scelta deriva dal fatto che, anche in ambito scolastico, ho avuto ampie possibilità di cimentarmi in questi campi e il mio desiderio era di mettermi alla prova in un progetto che richiedesse nuove competenze. D'altro canto il progetto di Project Manager non aveva attirato la mia attenzione. Il progetto di Pianificazione della produzione invece mi ha colpito sin da subito, richiedendo nuove competenze in ambito teorico, soprattutto per quanto riguarda tecniche Ricerca Operativa applicate ad algoritmi di pianificazione. Anche se è richiesto di ampliare e migliorare un applicativo già sviluppato ciò non mi ha creato problemi in quanto le nuove aggiunte sarebbero state atomiche dal punto di vista delle funzionalità e avrebbero portato ad un consistente miglioramento dell'applicativo una volta portato a termine il periodo di stage.

1.1.6 Convenzioni tipografiche

Vengono qui definite le regole tipografiche adottate nella stesura del testo:

- le abbreviazioni, gli acronimi e i termini che possono risultare ambigui o facenti parte del linguaggio tecnico o troppo specifico vengono definiti in modo approfondito alla fine del seguente documento in una sezione chiamata glossario;
- la prima occorrenza dei termini sopracitati viene identificata da un g a pedice come segue: glossario_G;

Capitolo 2

Descrizione dello stage

2.1 Panoramica del progetto

L'applicativo in questione è già in attività da circa un anno e ha il compito di pianificare la produzione settimanale degli ordini richiesti dalle varie aziende. Ogni azienda definisce un insieme di linee sulle quali è possibile produrre i prodotti e fornisce anche i giorni e gli orari di lavoro. Con queste informazioni il precedente programma era in grado di fornire una pianificazione distribuita sull'arco della settimana, valutando quale fosse l'ordinamento migliore e quali ordini scartare se il tempo a disposizione non fosse stato sufficiente. Questa logica di pianificazione ometteva però il controllo della disponibilità di materie prime e semilavorati, la quale veniva considerata sufficiente a produrre tutti gli ordini richiesti. Non veniva inoltre considerato il tempo di produzione degli eventuali semilavorati richiesti, e non si considerava nemmeno l'arrivo di materiali per il magazzino da parte dei fornitori. Quello che mi è stato richiesto dunque è l'integrazione delle parti sopracitate.

2.2 Specifiche tecniche del problema

Nel primo periodo di stage ho speso il tempo che avevo a disposizione studiando le componenti del problema che dovevo affrontare, ricavando un quadro generale del funzionamento logico della pianificazione della produzione. L'ostacolo principale si è rivelato essere la complessità in ambito lavorativo reale della pianificazione. Ho dovuto spendere del tempo assieme al tutor interno per riuscire ad entrare nel contesto in cui avrei dovuto lavorare, facendomi spiegare le varie sfaccettature e le scelte implementative prese. Di seguito voglio fornire un'idea del problema e di come avviene la pianificazione di un singolo prodotto finito. Perché un prodotto venga pianificato è necessario che ci sia almeno una linea in grado di produrlo, una volta definita la linea si deve scegliere una sequenza in cui produrre l'articolo. Una sequenza definisce giorno, data di inizio e di fine e può comprendere un insieme di vincoli da soddisfare (quali lavaggi oppure ordinaria manutenzione), l'articolo verrà quindi inserito all'interno della sequenza con il relativo tempo di produzione. Ogni linea ha solitamente più sequenze nelle quali è possibile collocare l'articolo. Ogni sequenza può avere dei vincoli di dipendenza con altre sequenze.

2.2.1 Implementazione

Segue la descrizione della struttura delle principali classi che vengono impiegate nell'applicativo.

Linee

Indica l'insieme delle linee sulle quali è consentita la produzione degli ordini, ogni linea ha relativo costo orario e definisce l'insieme degli articoli che può produrre con annessa velocità di produzione. Ogni linea è così definita:

- **Codice Linea:** serve ad indicare su quale linea si vuole produrre l'articolo;
- **Info Articolo:** lista che indica quali articoli la linea può produrre con relativo tempo di produzione;
- **Info Vincolo:** lista che indica i vincoli presenti sulla linea, possono essere obbligatori o opzionali;
- **Sequenze Linea:** lista che indica quali sono le sequenze di produzione della linea;
- **Pianificazione:** lista che indica quali articoli e quali vincoli sono stati pianificati nella linea corrente;
- **Errori:** indica quali errori si sono presentati durante la pianificazione;
- **Calendario:** indica quali sono i giorni lavorativi con le eventuali pause.

Sequenze

Indica l'insieme delle sequenze di produzione presenti su ogni linea, nelle quali vengono inseriti gli ordini che sono stati pianificati. Ogni sequenza è così definita:

- **Codice Sequenza:** serve ad indicare su quale sequenza si vuole inserire l'articolo;
- **Giorno:** indica il giorno nel quale si vuole inserire l'articolo;
- **Ora inizio/Ora fine:** indica gli orari di inizio e fine della sequenza corrente;
- **Elementi:** lista che indica gli articoli e i vincoli appartenenti alla sequenza corrente.

Ordine

Indica come si presenta un ordine da produrre. Ogni ordine è così definito:

- **Codice Articolo:** indica il codice dell'articolo;
- **Tipo Articolo:** indica il tipo dell'articolo, può essere un prodotto finito, semilavorato o materia prima;
- **Codice Linea:** indica il livello più generale della gerarchia di classificazione di un prodotto;
- **Codice Settore:** indica il terzo livello di gerarchia di classificazione;

- **Codice Famiglia:** indica il secondo livello di gerarchia, definisce la famiglia del prodotto;
- **Codice SottoFamiglia:** indica il livello più caratterizzante della classificazione di un prodotto;
- **Quantità:** indica la quantità richiesta da produrre;
- **Data Spedizione:** indica la data di spedizione dell'articolo;
- **Ora spedizione:** indica l'ora di spedizione dell'articolo;
- **Data Consegna:** indica la data di consegna presso la sede del cliente;
- **Linea preferenziale:** indica la linea preferenziale di produzione dell'articolo;
- **Anno Ordine:** indica l'anno dell'ordine in questione;
- **Materie Prime:** lista che indica l'insieme delle materie prime richieste per la produzione dell'ordine;
- **Semilavorati:** lista che indica l'insieme dei semilavorati richiesti per la produzione dell'ordine;
- **Riferimento Ordine:** lista che indica l'insieme degli ordini ai quali l'ordine corrente fa riferimento per la produzione di semilavorati;

2.2.2 Problemi logici da affrontare

Di seguito sono presentate le problematiche che l'algoritmo deve affrontare per ottenere una corretta pianificazione della produzione.

Temporalità

Qui vengono descritti tutti i problemi riguardanti i tempi di produzione.

- **Data Spedizione:** devono essere rispettate le date di spedizione e di consegna degli articoli da pianificare;
- **Tempo minimo alla consegna:** devono essere rispettate le date di inizio produzione nel caso di ordini con scadenza a breve termine;
- **Semilavorati:** i semilavorati di un ordine devono essere pianificati prima dell'ordine stesso;
- **Ordini Fornitori:** vanno considerate le date di arrivo delle materie prime da parte dei fornitori in modo da non scartare la produzione di ordini che sarebbero producibili;
- **Giorni Lavorativi:** devono essere rispettati i giorni lavorativi senza pianificare ordini al di fuori di essi;
- **Orario Lavorativo:** devono essere rispettati gli orari lavorativi senza eccedere dal monte ore impostato dall'azienda che esegue la pianificazione;
- **Sovrapposizione Sequenze:** ogni linea può produrre una singola sequenza per volta;
- **Sovrapposizione Articoli:** ogni sequenza può contenere un solo articolo senza sovrapporne degli altri nello stesso lasso di tempo.

Vincoli

Qui vengono descritti tutti i vincoli da rispettare.

- **Materie prime:** non è possibile produrre un ordine se non sono presenti sufficienti materie prime;
- **Vincoli Obbligatorii:** devono essere pianificati tutti i vincoli obbligatori di ogni sequenza;
- **Vincoli Condizionati:** devono essere pianificati tutti i vincoli condizionati di ogni sequenza in base alle condizioni imposte;
- **Vincoli Opzionali:** devono essere pianificati i vincoli opzionali solo in caso ci sia una finestra temporale sufficiente altrimenti si lascia spazio alla produzione di articoli;
- **Vincoli Articolo:** devono essere rispettati i vincoli di produzione di ogni articolo, quali linea preferenziale o linea con maggiore velocità di produzione.

Scelte implementative

Dopo una discussione col tutor interno abbiamo scartato l'ultimo vincolo facoltativo FA2, presente nel piano di lavoro, il quale riguardava la ripianificazione degli articoli in caso di guasti o necessità aziendali. Optando per una semplice nuova esecuzione dell'applicativo con i nuovi dati interessati.

2.3 Obiettivi

Gli obiettivi da raggiungere durante il progetto sono elencati di seguito, fanno riferimento a quanto riportato nella versione definitiva del piano di lavoro, con qualche aggiunta o modifica in sede di sviluppo, previa discussione con il tutor aziendale. Per ogni obiettivo è definito il grado di completamento: nullo, parziale, totale.

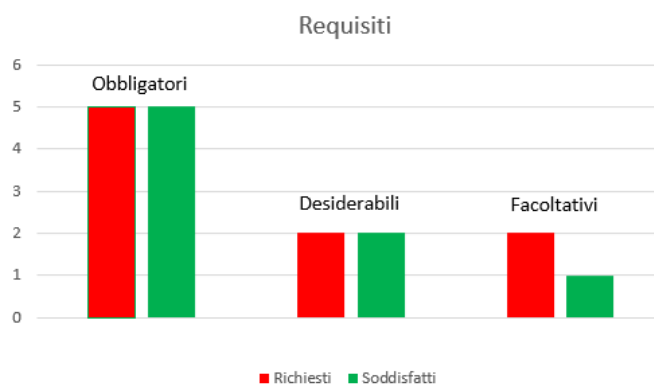


Figura 2.1: Requisiti

2.3.1 Requisiti obbligatori

- Ottimizzazione della pianificazione tenendo conto delle giacenze di magazzino: raggiungimento totale;
- Sviluppo di nuove strategie di scelta dell'Algoritmo Greedy e confronto dei risultati ottenuti in relazione alla funzione obiettivo: raggiungimento totale;
- Integrazione della Tabu Search con nuovi criteri di aspirazione e arresto: raggiungimento totale;
- Sviluppo di nuovi meccanismi di esplorazione del vicinato nella Tabu Search: raggiungimento totale;
- Acquisizione di competenze sull'utilizzo di algoritmi di Ricerca Operativa e applicazione in un caso di studio reale: raggiungimento totale.

2.3.2 Requisiti desiderabili

- Ottimizzazione della pianificazione tenendo conto degli ordini a fornitore presenti a sistema: raggiungimento totale;
- Ottimizzazione della pianificazione tenendo conto dei semilavorati: raggiungimento totale;

2.3.3 Requisiti opzionali

- Utilizzo di multithreading nelle fasi in cui è richiesta una maggiore capacità di calcolo: raggiungimento totale;
- Data una pianificazione inserita a sistema, eseguire una ripianificazione considerando vincoli dovuti a necessità aziendali dell'ultimo momento (anomalie, guasti, ordini urgenti, etc):raggiungimento nullo.

Come accennato nella sezione precedente il requisito FA2 "Data una pianificazione inserita a sistema, eseguire una ripianificazione considerando vincoli dovuti a necessità aziendali dell'ultimo momento" è stato scartato a seguito di una riunione con il tutor aziendale. Siamo quindi giunti alla conclusione che, in termini di efficacia, è preferibile una nuova esecuzione dell'algoritmo a partire da zero piuttosto che conservare gli attuali dati e procedere con una nuova rielaborazione. Questo rende anche più facile la definizione e l'aggiunta dei nuovi vincoli imposti, quali linee bloccate o sequenze non più attive, che potrebbero influenzare negativamente i precedenti dati inseriti.

2.4 Divisione settimanale

La seguente sezione vuole descrivere come sono state suddivise le 320 ore previste per il periodo di stage, affiancando ad ogni settimana lavorativa i requisiti e gli obiettivi raggiunti. Il periodo di stage inizia in data 09-settembre-2019 con termine ufficiale in data 01-novembre-2019 traslata al 03-novembre-2019 a causa del sostenimento di due prove d'esame.

- **Prima settimana:**

- analisi del modulo software esistente;
- funzionalità da realizzare;
- studio della documentazione disponibile dell'algoritmo esistente.
- **Seconda settimana:**
 - analisi dei rischi;
 - stesura dell'analisi dei requisiti che comprende le nuove funzionalità da integrare nel software esistente;
 - inizio della stesura di documentazione a supporto dell'architettura utilizzata.
- **Terza settimana:**
 - Studio delle tecnologie aziendali necessarie allo sviluppo del modulo in particolare linguaggio di programmazione VB.NET_G, componenti DevExpress_G e database Informix_G;
 - training_G sulle nuove tecnologie da utilizzare con realizzazione di un software di prova;
 - studio di algoritmi e tecniche di Ricerca Operativa e Ottimizzazione Combinatoria;
 - riunioni col tutor aziendale per decidere le scelte implementative dell'algoritmo greedy.
- **Quarta settimana:**
 - Algoritmo Greedy: sviluppo di nuove strategie di scelta del passo successivo adottato dall'algoritmo nella costruzione della soluzione del problema;
 - sviluppo procedura di gestione delle giacenze di magazzino.
- **Quinta settimana:**
 - sviluppo procedura di gestione dei semilavorati pianificati;
 - sviluppo procedura di gestione degli ordini fornitori.
- **Sesta settimana:**
 - integrazione della Tabu Search con nuovi meccanismi: criteri di aspirazione e arresto, variazione dei meccanismi di esplorazione del vicinato e adozione di tecniche di intensificazione e diversificazione.
- **Settima settimana:**
 - fase di test dei dati con valori reali forniti dai clienti di Ergon;
 - comparazione dell'algoritmo ultimato con il precedente algoritmo in uso;
 - verifica della bontà della soluzione in rapporto coi dati forniti dai clienti.
- **Ottava settimana:**
 - stesura della documentazione a supporto del prodotto sviluppato, con risalto sulle scelte implementative non banali.

Da sottolineare il fatto che le fasi di test sono state molteplici e non solo durante la settimana settimana. Ad ogni nuova aggiunta veniva effettuato un controllo sul risultato prodotto dalla pianificazione con dei dati reali di supporto, tutto ciò per garantire la correttezza della logica del codice in aggiunta. In comune accordo con il tutor abbiamo deciso di iniziare dalle parti più complesse la nuova fase di implementazione, così da garantire una continua verifica ad ogni aggiunta delle funzionalità meno importanti.

2.5 Analisi dei rischi

Ho trovato importante individuare eventuali rischi che possono portare a problematiche in grado di far procedere a rilento la realizzazione del progetto di stage. Di seguito viene presentata la tabella contenente i rischi preventivati durante la seconda settimana di stage. Ogni rischio possiede un codice identificativo, una breve descrizione affiancata dal suo rilevamento e relativo grado di rischio. Per ogni rischio è definito inoltre un piano di contingenza da seguire in caso di occorrenza del rischio.

Tabella 2.5.1: Tabella dei rischi di progetto

Codice Nome	Descrizione	Piano di contingenza	Grado di rischio
RO1 Problematiche di Ricerca Operativa	Dovendo svolgere uno studio individuale delle tematiche di Ricerca Operativa da affrontare, è possibile imbattersi in argomenti poco intuitivi o complicati da apprendere in solitaria.	Vengono fornite le dispense riguardanti il contesto in cui si andrà a lavorare e in caso di incomprensioni si farà affidamento sia sul tutor interno sia al proprio relatore.	Occorrenza: Media Pericolosità: Alta
RT1 Inesperienza tecnologica	Alcune delle tecnologie adottate sono nuove, pertanto è possibile incorrere in problemi durante lo svolgimento delle attività che le coinvolgono.	Viene fornita la documentazione ufficiale in modo da avere ampio supporto per qualsiasi lacuna di natura tecnica. Se ciò non dovesse bastare si farà affidamento al tutor interno o qualche membro delegato del team di sviluppo di Ergon.	Occorrenza: Media Pericolosità: Media
RT2 Scelte imple- mentative	Non è detto che tutte le scelte prese in considerazione durante lo studio del problema portino ad una corretta soluzione o ad una esecuzione in tempi accettabili.	Verranno valutate, in caso di necessità, nuove strade da percorrere per la risoluzione del problema.	Occorrenza: Bassa Pericolosità: Alta

Capitolo 3

Progetto di stage

3.1 Analisi dei requisiti

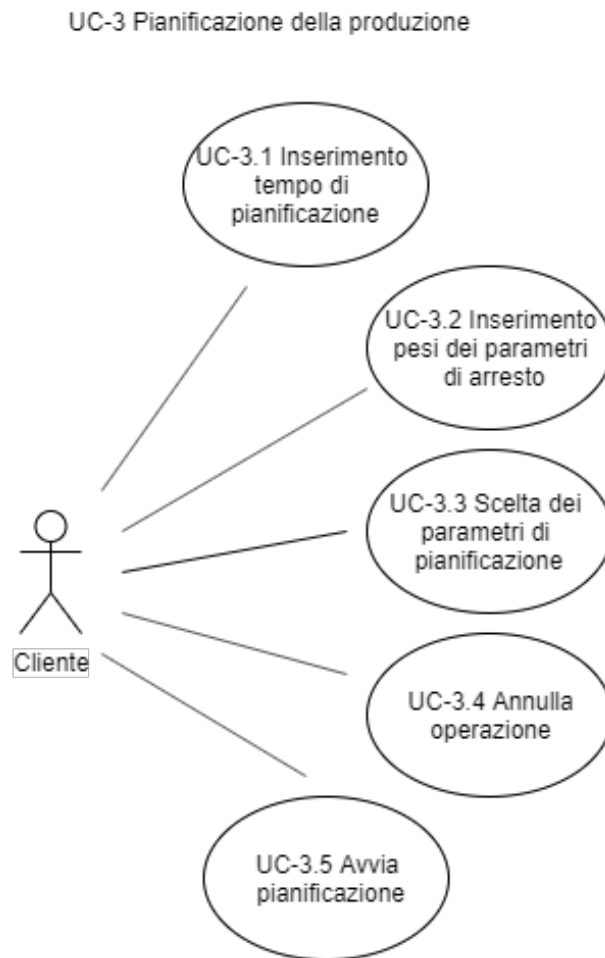
Prima di iniziare a lavorare sul progetto, il tutor aziendale, mi ha fornito la documentazione esistente dell'applicativo, in particolar modo ho fatto riferimento al manuale utente creato in precedenza e all'analisi dei requisiti. Mi è stato espressamente chiesto inoltre di non eseguire alcuna modifica a livello grafico dell'applicativo in quanto la clientela era abituata alla versione corrente e non sempre è immediato adattarsi a cambiamenti di questo tipo. Ho infine aggiunto due casi particolari che verranno evidenziati di seguito per quanto riguarda l'analisi dei requisiti esistente.

3.1.1 Casi d'uso

Per rappresentare l'insieme delle azioni comuni ad un singolo utente sono stati utilizzati i casi d'uso. Per descrivere i vari scenari ho mantenuto la precedente scaletta di informazioni così da dare una continuità al documento esistente. Ciascun caso d'uso sarà quindi rappresentato come descritto di seguito:

- **Identificatore** univo nel formato UC-[Codice];
- **Titolo** del caso d'uso;
- **Descrizione** generale del caso d'uso;
- **Attori** coinvolti: primari e secondari;
- **Precondizione** del caso d'uso;
- **Scenario principale** che il caso d'uso vuole modellare, identificando ogni azione che ne fa parte;
- **Postcondizione** del caso d'uso;
- **Estensioni** dello scenario principale, evidenziata da una lettera maiuscola e inserita in un elenco.

Di seguito vengono rappresentate le modifiche apportate al documento esistente. Si fa riferimento allo standard UML 2.0 per la rappresentazione dei casi d'uso.

UC-3 Pianificazione della produzione**Figura 3.1:** Use Case - UC-3

Attori: Cliente.

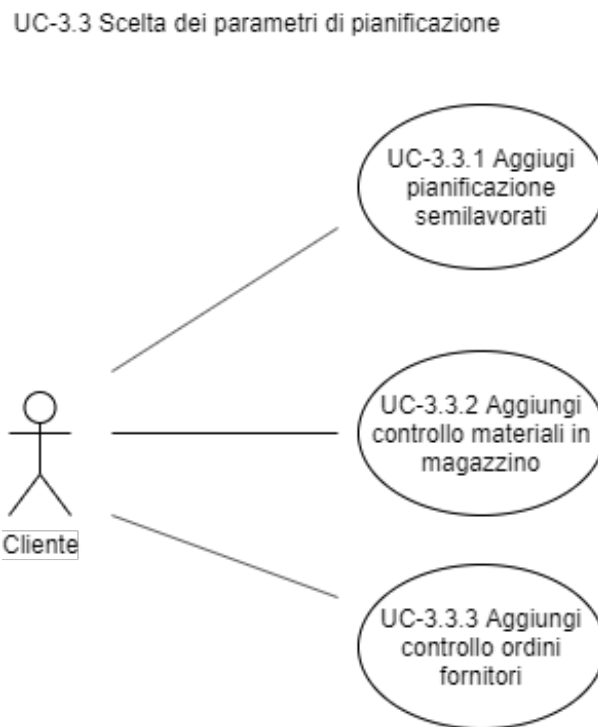
Descrizione: Il cliente esegue la pianificazione degli ordini selezionati.

Precondizione: Il cliente ha selezionato gli ordini che intende pianificare e i giorni in cui vuole farlo.

Scenario principale:

- Il cliente seleziona che ordini vuole pianificare;
- Il cliente preme il pulsante "Pianifica";
- Il cliente modifica i parametri della pianificazione come previsto dagli UC-3.1, UC-3.2, UC-3.3;
- Il cliente preme il pulsante "Avvia pianificazione".

Postcondizione: Nella finestra principale viene visualizzato il risultato della pianificazione.

UC-3.3 Scelta dei parametri di pianificazione**Figura 3.2:** Use Case - UC-3.3

Attori: Cliente.

Descrizione: Il cliente seglie i parametri da considerare nella pianificazione.

Precondizione: Il cliente ha selezionato il pulsante "Metodo di pianificazione".

Scenario principale:

- Il cliente seleziona come effettuare le pianificazione;
- Il cliente può selezionare le varie aggiunte descritte dagli UC-3.3.1, UC-3.3.2, UC3.3.3;
- Il cliente preme il pulsante "Conferma".

Postcondizione: Si verrà ricondotti allo use case UC-3.3.

3.2 Tracciamento dei requisiti

Ogni requisito è composto dalla seguente struttura:

- **codice identificativo:** ogni codice identificativo è univoco e conforme alla seguente codifica:

R[Importanza][Tipologia][Codice]

Il significato delle cui voci è:

- **Tipologia:** ogni requisito può assumere uno dei seguenti valori:
 - * *F*: funzionale;
 - * *P*: prestazionale;
 - * *Q*: qualitativo;
 - * *V*: vincolo.
 - **Importanza:** ogni requisito può assumere uno dei seguenti valori:
 - * *O*: requisito obbligatorio: irrinunciabili per qualcuno degli stakeholder;
 - * *D*: requisito desiderabile: non strettamente necessari ma a valore aggiunto riconoscibile;
 - * *F*: requisito facoltativo: relativamente utili oppure contrattabili più avanti nel progetto.
 - **Codice:** è un identificatore univoco del requisito segue un ordine incrementale.
- **classificazione:** viene riportata l'importanza del requisito. Sebbene questa sia un'informazione ridondante ne facilita la lettura;
 - **descrizione:** descrizione breve ma completa del requisito, meno ambigua possibile.

3.2.1 Requisiti funzionali

Tabella 3.2.1.1: Tabella dei requisiti funzionali

Requisito	Classificazione	Descrizione
RFO1	Obbligatorio	Il sistema permette l'inserimento di un nuovovincolo di linea
RFO2	Obbligatorio	Il sistema permette la modifica dei dati di unvincolo di linea esistente
RFO3	Obbligatorio	Il sistema permette l'eliminazione di un vincolo dilinea
RFO4	Obbligatorio	L'interfaccia permette l'eliminazione di una linea
RFO5	Obbligatorio	Il sistema permette l'aggiunta di una nuovasequenza
RFO6	Obbligatorio	Il sistema permette la modifica dei dati di unasequenza esistente
RFO7	Obbligatorio	Il sistema permette l'eliminazione di un articolo disequenza esistente

Tabella 3.2.1.1: (continua)

Requisito	Classificazione	Descrizione
RFO8	Obbligatorio	Il sistema permette l'eliminazione di un vincolo disequenza esistente
RFO9	Obbligatorio	Il sistema permette l'eliminazione di una sequenza
RFO10	Obbligatorio	Il sistema permette all'utente di scegliere un tempomassimo di esecuzione dell'algoritmo
RFO11	Obbligatorio	Il sistema permette all'utente di assegnare un pesoall'importanza di parametri di produzione quali:pezzi prodotti e occupazione delle linee
RFD1	Desiderabile	Il sistema permette all'utente di scegliere sericalcolare la pianificazione già esistente
RFO12	Obbligatorio	Il sistema permette di pianificare automaticamente la produzione dato un insieme di ordini e unintervallo di tempo
RFO1	Obbligatorio	Il sistema permette di scegliere i relativi criteri di pianificazione

3.2.2 Requisiti qualitativi

Tabella 3.2.2.1: Tabella dei requisiti qualitativi

Requisito	Classificazione	Descrizione
RQD1	Desiderabile	Il programma genera un file di Log_G di supporto al Debug_G
RQO1	Obbligatorio	Ogni scelta non banale effettuata è adeguatamentecommentata nel codice
RQF1	Facoltativo	Fornire un manuale utente

Tabella 3.2.2.1: (continua)

Requisito	Classificazione	Descrizione
RQF2	Facoltativo	Fornire un manuale sviluppatore

3.2.3 Requisiti prestazionali

Tabella 3.2.3.1: Tabella dei requisiti prestazionali

Requisito	Classificazione	Descrizione
RPD1	Desiderabile	L'applicativo deve fornire una soluzione entro il tempo stabilito dall'utente
RPD2	Desiderabile	L'applicativo deve fornire la soluzione nel modo più rapido possibile senza valutare ulteriori combinazioni se è stato raggiunto un determinato punteggio stabilito per la soluzione stessa
RPD3	Desiderabile	La fase di lettura dei dati deve essere eseguita in tempi inferiori ai 30 minuti

3.2.4 Requisiti vincolo

Tabella 3.2.4.1: Tabella dei requisiti prestazionali

Requisito	Classificazione	Descrizione
RVO1	Desiderabile	Rendere compatibile il programma con l'attuale modulo di pianificazione della produzione
RVO2	Desiderabile	Ottimizzare la pianificazione tenendo conto delle materie prime e dei semilavorati

Tabella 3.2.4.1: (continua)

Requisito	Classificazione	Descrizione
RVO3	Desiderabile	Ottimizzare la pianificazione tenendo conto degli ordini dei fornitori inseriti a sistema
RVO4	Desiderabile	Ottimizzare la pianificazione rendendo possibile la pianificazione dei semilavorati
RVO5	Desiderabile	Il programma è sviluppato tramite il linguaggio di programmazione Vb.NET v.4.7.0
RVO6	Desiderabile	L'IDE _G utilizzato è Microsoft Visual Studio _G v.10.0.4
RVO7	Desiderabile	Le componenti grafiche sono basate sulle DevEx-press _G
RVO8	Desiderabile	I dati vengono salvati su un database Informix
RVO9	Desiderabile	I dati in input sono scritti su un file JSON
RVO10	Desiderabile	I dati in output sono scritti su un file JSON

3.2.5 Riepilogo dei requisiti

Tabella 3.2.5.1: Tabella del riepilogo dei requisiti

Tipo	Obbligatorie		Desiderabili	Facoltative
Funzionali	13	1		0
Qualitativi	1	1		2
Prestazionali	0	3		0
Di vincolo	10	0		0
Totali	24	5		2

3.3 Tecnologie e strumenti

Di seguito sono riportate tutte le tecnologie utilizzate durante lo sviluppo del progetto. Tali scelte sono state imposte da Ergon Informatica in quanto sono gli strumenti che vengono utilizzati dall'azienda per lo sviluppo dei progetti interni. La scelta è inoltre guidata dal fatto che, per mantenere l'integrazione con le parti già esistenti dell'applicativo e il suo funzionamento tramite il software Ergdis, non si è potuto spostarsi su nuove tecnologie, magari anche solo a qualche versione successiva di esse. Ergon Informatica si appoggia a questi strumenti in quanto sono forniti di un'ottima documentazione di supporto, hanno un alto tasso di scalabilità e forniscono un supporto in tempo reale alla codifica.

Vb.NET



Figura 3.3: Logo Vb.NET

Linguaggio di programmazione che deriva da Visual Basic 6, è sviluppato da Microsoft e, al contrario dei precedenti, supporta il paradigma di programmazione orientata agli oggetti. Vb.NET consente lo sviluppo di applicazioni Windows, Web e per dispositivi mobili. Come avviene con tutti i linguaggi basati su Microsoft .NET Framework, i programmi scritti in Vb.NET usufruiscono delle funzionalità di sicurezza e interoperabilità dei linguaggi. Nel progetto viene utilizzato questo linguaggio per poter usufruire delle classi, le quali riducono significativamente le duplicazioni di codice e rendono l'intero sorgente chiuso alle modifiche e aperto agli ampliamenti. Viene inoltre utilizzato in quanto è il linguaggio attualmente adottato dall'azienda per lo sviluppo e si presta bene al tipo di lavorazioni necessarie a portare a compimento lo sviluppo dell'applicativo.

Visual Studio 2010



Figura 3.4: Logo Visuali Studio 2010

Ambiente di sviluppo integrato sviluppato da Microsoft. La prima versione risale al 1997 e aveva lo scopo di fornire un ambiente di sviluppo grafico ed integrato che aiutasse lo sviluppatore a gestire i progetti in maniera semplice, ma efficace, aumentandone quindi la produttività. Microsoft ha incluso il supporto a differenti linguaggi di programmazione. In Ergon Informatica è il principale ambiente di sviluppo sia per applicazioni desktop che mobile, di conseguenza è stato utilizzato anche durante lo sviluppo del progetto.

DevExpress



Figura 3.5: Logo DevExpress

Developer Express Inc. è una società di sviluppo software fondata nel 1998, inizialmente fornisce un insieme di controlli UI_G poi sviluppa diverse estensioni per le librerie grafiche. In particolare nel progetto ci si è appoggiati a questa tecnologia per velocizzare la creazione dell'interfaccia grafica sulla quale si basa il modo di rappresentare la soluzione che viene fornita, permettendo una chiara visualizzazione tabellare grazie ad una delle tante estensioni $_G$ utilizzabili.

JSON

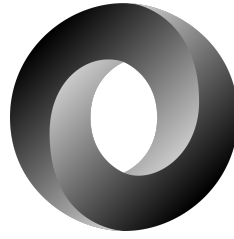


Figura 3.6: Logo JSON

Acronimo di *JavaScript Object Notation*, è un formato adatto all'interscambio di dati fra applicazioni client/server_G. Serve, in particolare, a fornire una struttura a dati interessati rendendoli interscambiabili tra diverse applicazioni senza che si renda necessaria la codifica e decodifica di quest'ultimi. Nel nostro caso si rende utile quando è necessario delegare l'esecuzione di operazioni sui dati prelevati da un terminale con accesso al database verso un terminale esterno. I dati raccolti dal database venivano trasferiti in un file JSON, il quale veniva inviato al successivo passo di esecuzione dell'algoritmo, al termine di ciò veniva restituita la soluzione sempre in formato JSON a chi ne aveva fatto richiesta e di conseguenza visualizzata.

Informix



Figura 3.7: Logo Informix

Fa parte della divisione DBMS_G di IBM_G, è un sistema software progettato per consentire la creazione, la manipolazione e l'interrogazione efficiente di database. L'Informix server supporta il modello relazionale ad oggetti che permette ad IBM di offrire estensioni che supportano i tipi di dati che non sono una parte dello standard SQL_G. Le estensioni più usate sono quelle riguardanti le serie temporali e spaziale che forniscono entrambe supporto al tipo di dato ed estensioni linguistiche che permettono interrogazioni per un dominio specifico ad alte prestazioni e archiviazione efficiente per set di dati basati su serie temporali e dati spaziali. È il servizio database sul quale fa affidamento l'azienda essendo anche uno dei partner principali.

Capitolo 4

Progettazione e sviluppo

4.1 Progettazione

4.1.1 Architettura

Prima di descrivere le varie aggiunte che sono state apportate al software è necessario fornire un'idea dell'architettura sulla quale il progetto si basa. In questo modo si rendono più chiare molte delle scelte che sono state prese e il perchè altre sono state scartate. Di seguito è presentato uno schema ad alto livello di come sono strutturate le varie componenti che formano l'intero sistema. Come possiamo notare dalla figura sottostante, il progetto segue un flusso pressochè "ciclico", partendo dal recupero dei dati dal database e infine fornendo la soluzione. I dati estratti dal database vengono convertiti in file JSON il quale verrà letto dall'algoritmo che ha il compito di trovare la soluzione, una volta fornita la soluzione essa viene ritradotta in un file JSON il quale viene letto e trasposto nell'interfaccia utente.

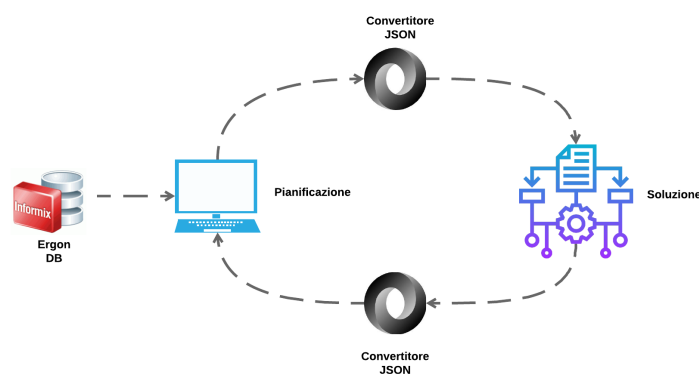


Figura 4.1: Architettura generale

4.1.2 Algoritmo

Il problema

Dovendo integrare delle nuove funzionalità al programma esistente, si è prima effettuata un'analisi del problema esistente per quanto riguarda la pianificazione della produzione, in modo da poter comprendere alcune delle scelte euristiche $_G$ che sono state effettuate in seguito. Il suddetto problema rientra nella categoria NP-hard $_G$ in quanto è riconducibile al problema del commesso viaggiatore $_G$. A tali problemi non è possibile far corrispondere una soluzione ottima ma, al massimo, la miglior approssimazione possibile di una soluzione ammissibile. In aggiunta al problema del commesso viaggiatore si devono considerare altri vincoli come:

- tempo massimo sul quale si vogliono pianificare gli ordini;
- numero di linee sulle quali si possono produrre determinati ordini;
- vincoli generali di sequenza che andremo a definire in seguito.

Questo è l'insieme di problematiche riguardanti la precedente fase del progetto, dove rientrava solo la pianificazione dei prodotti finiti. Con la richiesta di estendere tale progetto, il numero di vincoli da soddisfare cresce in proporzione al numero di ordini che si vogliono pianificare. In particolare vanno tenuti in considerazione i vincoli precedenti più l'aggiunta dei seguenti:

- pianificazione dei semilavorati antecedente ai prodotti finiti;
- materie prime e semilavorati presenti in quantità sufficienti per la produzione;
- verifica della disponibilità di materie prime e semilavorati in base agli ordini fornitori.

In conclusione, il problema che il progetto si pone di risolvere, consiste nel trovare una soluzione ammissibile, quanto più vicina all'ottimo, della pianificazione in base ai vincoli imposti.

La soluzione

Dovendo appoggiarsi all'algoritmo esistente, non è stato possibile eseguire uno stravolgimento completo del suo funzionamento, anzi si è cercato di inserire le nuove funzionalità in modo da integrarle il più possibile senza effettuare modifiche sostanziali alla struttura del codice esistente. Per fare ciò si è fatto ampio uso di nuove funzioni e classi, per poter delegare, dalla base esistente, l'esecuzione di nuovi moduli di pianificazione. La struttura di partenza sulla quale vengono integrati i nuovi moduli è composta da un'iniziale fase di lettura dei dati dal database. Tali dati vengono convertiti in un file JSON che, per comodità di esecuzione, viene letto nuovamente dallo stesso programma, anche se l'idea di base sarebbe quella di delegare la sua lettura e successiva elaborazione dei dati ad un sistema centralizzato esterno all'azienda che fornisce i dati stessi. Una volta ottenuti i dati in formato JSON, vengono inseriti nell'algoritmo tramite l'impiego di classi e strutture predisposte. Tali classi vengono utilizzate come parametri per la prima funzione che si pone come obiettivo quello di fornire una soluzione ammissibile nel minor tempo possibile ovvero l'Algoritmo Greedy $_G$. Tale algoritmo ha il compito di valutare, in modo sequenziale, i vari ordini che necessitano di essere pianificati, se un ordine soddisfa i vincoli imposti allora verrà inserito nella soluzione iniziale. Una volta ottenuta la soluzione iniziale, questa

viene fornita alla successiva funzione di ottimizzazione, composta da un algoritmo di Tabu Search_G. Tale algoritmo ha il compito di eseguire un insieme di mosse specifiche in modo casuale, valutando di volta in volta la bontà della soluzione ottenuta dopo la sua esecuzione. Al termine di un numero prefissato di iterazioni, o dopo aver soddisfatto i criteri di aspirazione, verrà fornita la soluzione ottimizzata da visualizzare. Visto che le scelte implementative non sempre possono essere chiare o si potrebbe pensare di utilizzare una metodologia diversa, viene illustrato di seguito il funzionamento degli algoritmi su cui fa affidamento la pianificazione della produzione.

Algoritmo Greedy

È il primo passo per ottenere una soluzione ammissibile del problema in questione. Tale algoritmo si sviluppa tenendo conto che non è possibile effettuare dei "passi indietro", ovvero, una volta inserito un ordine in pianificazione tale ordine rimane fino alla fine dell'esecuzione. I parametri in ingresso di tale algoritmo sono i seguenti:

- ordini da pianificare;
- materie prime e semilavorati presenti in magazzino;
- materie prime e semilavorati presenti in ordini fornitori;
- linee di lavorazione con i rispettivi vincoli.

L'ordine di funzionamento è quindi il seguente:

- prima scelta euristica, vengono ordinati i prodotti in base alla loro data di spedizione;
- il primo ordine in questione viene valutato;
- si verifica la presenza di materie prime, e qui abbiamo i seguenti casi:
 - le materie prime sono disponibili per produrre l'ordine, inserisco in pianificazione;
 - le materie prime non sono disponibili per produrre l'ordine, verifico l'arrivo di eventuali materie prime da parte dei fornitori, se ciò avviene inserisco in pianificazione;
 - non pianifico l'ordine.
- si verifica la presenza di semilavorati, e qui abbiamo i seguenti casi:
 - i semilavorati sono disponibili per produrre l'ordine, inserisco in pianificazione;
 - i semilavorati non sono disponibili per produrre l'ordine, provo ad eseguire la pianificazione dei semilavorati mancanti, se ciò avviene inserisco in pianificazione;
 - non pianifico l'ordine.
- ogni valutazione effettua un controllo se è possibile inserire l'ordine in base al tempo lavorativo rimanente;
- si ripetono le azioni sopracitate fino al termine della valutazione di tutti gli ordini da pianificare.

Si ricorda che non è necessario che tutti gli ordini richiesti siano pianificati in quanto uno dei precedenti vincoli può non essere soddisfatto e, se tale ordine venisse inserito nella soluzione, quest'ultima sarebbe non ammissibile perché viola appunto almeno uno dei vincoli imposti. Inoltre, se si esegue un accurato controllo, si può notare che il risultato fornito al termine della sua esecuzione, in alcuni casi, ha un ampio margine di miglioramento. L'algoritmo Greedy ha come obiettivo quello di creare una soluzione iniziale nel modo più rapido possibile, in modo da avere una base di partenza per la successiva ottimizzazione eseguita dall'algoritmo Tabu Search.

Capitolo 5

Progettazione e codifica

Breve introduzione al capitolo

5.1 Tecnologie e strumenti

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

Tecnologia 1

Descrizione Tecnologia 1.

Tecnologia 2

Descrizione Tecnologia 2

5.2 Ciclo di vita del software

5.3 Progettazione

Namespace 1

Descrizione namespace 1.

Classe 1: Descrizione classe 1

Classe 2: Descrizione classe 2

5.4 Design Pattern utilizzati

5.5 Codifica

Capitolo 6

Verifica e validazione

Capitolo 7

Conclusioni

7.1 Consuntivo finale

7.2 Raggiungimento degli obiettivi

7.3 Conoscenze acquisite

7.4 Valutazione personale

Appendice A

Appendice A

Citazione

Autore della citazione

Bibliografia