

Università degli Studi di Padova

DIPARTIMENTO DI MATEMATICA "TULLIO LEVI-CIVITA"

CORSO DI LAUREA IN INFORMATICA



Sviluppo di un modulo software per la gestione  
degli ordini di acquisto con l'utilizzo di metodi  
euristici di ottimizzazione

*Tesi di laurea*

*Relatore*

Prof. Luigi De Giovanni

*Laureando*

Filippo Brugnolaro

Matricola 1217321



*La disumanità del computer sta nel fatto che,  
una volta programmato e messo in funzione,  
si comporta in maniera perfettamente onesta.*

— Isaac Asimov

# Ringraziamenti

*In primis vorrei esprimere la mia gratitudine al Professor Luigi De Giovanni, relatore della mia tesi, per la disponibilità e l'aiuto fornitomi durante la stesura.*

*Desidero ringraziare con affetto la mia famiglia per tutto il sostegno e la vicinanza dimostrata in ogni momento e per non avermi mai fatto mancare nulla durante gli anni di studio.*

*Vorrei ringraziare i miei amici che mi sono stati vicini e mi hanno accompagnato in questi anni, soprattutto nei momenti difficili.*

*Infine desidero ringraziare in maniera speciale il mio amico Alessandro, che mi ha reso lo studio meno faticoso e con cui ho passato dei bei momenti, e Linpeng, che mi ha pazientemente guidato all'inizio del corso di laurea.*

*Padova, Settembre 2022*

Filippo Brugnolaro



# Indice

<b>1</b>	<b>Introduzione</b>	<b>1</b>
1.1	L'azienda . . . . .	1
1.2	L'idea . . . . .	2
1.3	Descrizione dello stage . . . . .	2
1.3.1	Introduzione . . . . .	2
1.3.2	Obiettivi . . . . .	2
1.3.3	Analisi preventiva dei rischi . . . . .	2
1.4	Organizzazione del testo . . . . .	3
<b>2</b>	<b>Studio di fattibilità</b>	<b>5</b>
2.1	Introduzione allo studio . . . . .	5
2.2	Soluzioni proposte . . . . .	5
2.2.1	Algoritmo Greedy . . . . .	5
2.2.2	Algoritmo Tabu Search . . . . .	5
2.2.3	Algoritmo Genetico . . . . .	5
2.3	Conclusioni dello studio . . . . .	5
<b>3</b>	<b>Analisi dei requisiti</b>	<b>7</b>
3.1	Casi d'uso . . . . .	7
3.2	Tracciamento dei requisiti . . . . .	8
<b>4</b>	<b>Progettazione e codifica</b>	<b>11</b>
4.1	Tecnologie e strumenti . . . . .	11
4.2	Progettazione . . . . .	11
4.3	Design Pattern utilizzati . . . . .	11
4.4	Codifica . . . . .	11
<b>5</b>	<b>Verifica e validazione</b>	<b>13</b>
5.1	Verifica . . . . .	13
5.1.1	Documentazione . . . . .	13
5.1.2	Testing del modulo . . . . .	13
5.2	Validazione . . . . .	13
5.2.1	Documentazione . . . . .	13
5.2.2	Codice . . . . .	13
<b>6</b>	<b>Conclusioni</b>	<b>15</b>
6.1	Prodotto finale . . . . .	15
6.2	Raggiungimento degli obiettivi . . . . .	15
6.3	Conoscenze acquisite . . . . .	15
<b>A</b>	<b>Appendice A</b>	<b>17</b>
	<b>Bibliografia</b>	<b>21</b>

# Elenco delle figure

1.1	Logo Ergon Informatica S.R.L. . . . .	<a href="#">1</a>
3.1	Use Case - UC0: Scenario principale . . . . .	<a href="#">7</a>

# Elenco delle tabelle

3.1	Tabella del tracciamento dei requisiti funzionali . . . . .	<a href="#">9</a>
3.2	Tabella del tracciamento dei requisiti qualitativi . . . . .	<a href="#">9</a>
3.3	Tabella del tracciamento dei requisiti di vincolo . . . . .	<a href="#">9</a>

# Capitolo 1

## Introduzione

*Nel seguente capitolo si introduce brevemente l'azienda ospitante e il progetto affrontato.*

### 1.1 L'azienda

Ergon Informatica S.R.L.<sup>1</sup>(da qui in poi "*Ergon*") è un'azienda italiana, fondata nel 1988, situata a Castelfranco Veneto.

Essa si occupa principalmente soluzioni gestionali per piccole e medie imprese e dello sviluppo di *software* **ERP** per i settori dell'alimentare e dei trasporti, ma completa l'offerta con la vendita di prodotti hardware, servizi web e hosting, nonché con progetti di *server consolidation* e virtualizzazione di sistemi. L'azienda inoltre si è sviluppata in maniera costante negli anni e oggi può vantare una posizione di tutto rispetto tra le aziende dello stesso settore. Attualmente fanno parte della stessa gestione:

- \* *Ergon Informatica S.R.L.*: che si occupa del *software*;
- \* *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 altre due parti.

Il logo dell'azienda è illustrato in [Figura 1.1](#).



**Figura 1.1:** Logo Ergon Informatica S.R.L.

Il prodotto proprietario dell'azienda è *ERGDIS*, sistema **ERP** il cui insieme dei moduli copre ogni aspetto della conduzione aziendale.

---

<sup>1</sup>Sito ufficiale: <https://www.ergon.it/>

In particolare vengono gestiti vari compiti che si dislocano dall'area amministrativa al controllo direzionale, dall'area commerciale alla pianificazione e al controllo della produzione, dalla gestione acquisti alla logistica di magazzino, all'archiviazione ottica alla gestione della qualità.

Alcuni di essi, inoltre, si possono interfacciare con dispositivi automatici presenti in azienda, come, ad esempio, linee di confezionamento o robot.

Paragrafo riguardo alle funzionalità di ERGDIS

Paragrafo riguardo agli obiettivi generali dell'azienda

## 1.2 L'idea

Lo *stage* proposto consiste nella progettazione e nello sviluppo di un modulo *software* volto ad assistere l'azienda nella fase di approvvigionamento dei prodotti dai propri fornitori, supportandola nel scegliere da quale fornitore e quando acquistare i prodotti.

Questa nuova funzionalità andrebbe ad ottimizzare un modulo già esistente che però, per ogni prodotto da ordinare, prende in considerazione l'ultima data d'ordine disponibile prima dell'inizio dell'effettiva copertura del fabbisogno del prodotto stesso. Questo dunque non garantirebbe con certezza una scelta ottimale in relazione alle possibilità d'ordine fornite dagli appositi listini e calendario dei fornitori.

Data la natura combinatoria del problema, il modulo dovrà fornire in tempi ragionevoli una "buona soluzione" del problema, ovvero tendente il più possibile all'ottimo, e dovrà integrarsi con l'intero sistema *ERGDIS*.

È previsto inoltre che i dati su cui si è fatta l'ottimizzazione e il confronto dei risultati vengano visualizzati tramite un'apposita interfaccia grafica che verrà sviluppata in linea con l'ambiente di sviluppo dell'azienda ([.NET Framework](#) e [DevExpress](#)).

## 1.3 Descrizione dello stage

### 1.3.1 Introduzione

### 1.3.2 Obiettivi

Di seguito vengono elencati tutti gli obiettivi previsti dallo *stage*:

- \* Analisi del contesto *ERGDIS*;
- \* Studio dei principali algoritmi di ricerca operativa e ottimizzazione combinatoria;
- \* Redazione di uno studio di fattibilità con integrazione di micro-moduli di test;
- \* Redazione di un'analisi dei requisiti;
- \* Sviluppo e codifica del modulo *software* con le tecnologie utilizzate dall'azienda;
- \* Redazione di documentazione tecnica riguardante le scelte implementative e architetturali effettuate;
- \* *Report* finale sui risultati ottenuti.

### 1.3.3 Analisi preventiva dei rischi



## 1.4 Organizzazione del testo

Di seguito viene illustrata l'organizzazione dei capitoli successivi:

**Il secondo capitolo** approfondisce lo studio di fattibilità effettuato, utile per entrare a conoscenza delle più utilizzate tecniche di ottimizzazione combinatoria e per analizzare quali siano i vantaggi e svantaggi di ognuno di essi.

**Il terzo capitolo** descrive l'analisi dei requisiti del progetto, comprensiva di diagrammi dei casi d'uso e raccolta dei requisiti derivanti dall'analisi di questi ultimi.

**Il quarto capitolo** approfondisce le fasi di progettazione e codifica, comprensiva di diagrammi delle classi e di approfondimenti a livello implementativo.

**Il quinto capitolo** espone tutte verifiche effettuate durante il progetto e la validazione finale a conferma dei requisiti inizialmente stilati nella fase di [analisi dei requisiti](#).

**Il sesto capitolo** presenta le conclusioni tratte dallo *stage*, comprensivo di conoscenze acquisite e considerazioni di carattere personale.

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

- \* gli acronimi, le abbreviazioni e i termini ambigui o di uso non comune menzionati vengono definiti nel glossario, situato alla fine del presente documento;
- \* i termini in lingua straniera o facenti parti del gergo tecnico sono evidenziati con il carattere *corsivo*.



## Capitolo 2

# Studio di fattibilità

*Brevissima introduzione al capitolo*

### 2.1 Introduzione allo studio

### 2.2 Soluzioni proposte

#### 2.2.1 Algoritmo Greedy

#### 2.2.2 Algoritmo Tabu Search

#### 2.2.3 Algoritmo Genetico

### 2.3 Conclusioni dello studio



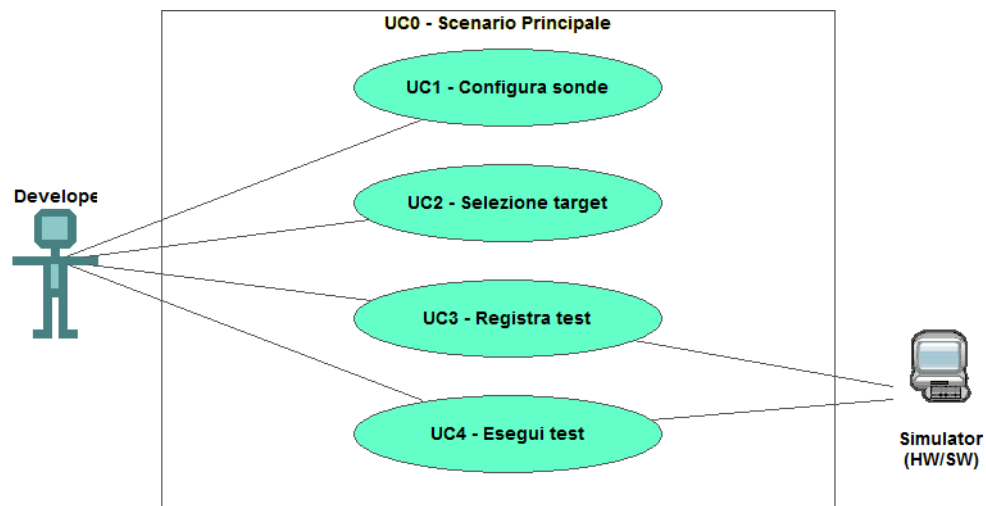
## Capitolo 3

# Analisi dei requisiti

*Breve introduzione al capitolo*

### 3.1 Casi d'uso

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



**Figura 3.1:** Use Case - UC0: Scenario principale

#### UC0: Scenario principale

**Attori principali:** Sviluppatore applicativi.

**Pre-condizione:** Lo sviluppatore è entrato nel plug-in di simulazione all'interno dell'IDE.

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

**Post-condizione:** Il sistema è pronto per permettere una nuova interazione.

## 3.2 Tracciamento dei requisiti

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

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

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

R = requisito

F = funzionale

Q = qualitativo

V = di vincolo

N = obbligatorio (necessario)

D = desiderabile

Z = opzionale

Nelle tabelle [3.1](#), [3.2](#) e [3.3](#) sono riassunti i requisiti e il loro tracciamento con gli use case delineati in fase di analisi.

**Tabella 3.1:** Tabella del tracciamento dei requisiti funzionali

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

**Tabella 3.2:** Tabella del tracciamento dei requisiti qualitativi

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

**Tabella 3.3:** Tabella del tracciamento dei requisiti di vincolo

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





## Capitolo 4

# Progettazione e codifica

*Breve introduzione al capitolo*

### 4.1 Tecnologie e strumenti

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

#### **Tecnologia 1**

Descrizione Tecnologia 1.

#### **Tecnologia 2**

Descrizione Tecnologia 2

### 4.2 Progettazione

#### **Namespace 1**

Descrizione namespace 1.

**Classe 1:** Descrizione classe 1

**Classe 2:** Descrizione classe 2

### 4.3 Design Pattern utilizzati

### 4.4 Codifica



## Capitolo 5

# Verifica e validazione

### 5.1 Verifica

#### 5.1.1 Documentazione

#### 5.1.2 Testing del modulo

### 5.2 Validazione

#### 5.2.1 Documentazione

#### 5.2.2 Codice



## Capitolo 6

# Conclusioni

6.1 Prodotto finale

6.2 Raggiungimento degli obiettivi

6.3 Conoscenze acquisite



Appendice A

Appendice A

Citazione

---

Autore della citazione









# Bibliografia

## Riferimenti bibliografici

James P. Womack, Daniel T. Jones. *Lean Thinking, Second Edition*. Simon & Schuster, Inc., 2010.

## Siti web consultati

*Manifesto Agile*. URL: <http://agilemanifesto.org/iso/it/>.