



UNIVERSITÀ DEGLI STUDI DI PADOVA  
CORSO DI LAUREA IN INFORMATICA (L-31)

Corso di Ingegneria del Software  
Anno Accademico 2025/2026

# Analisi dei Requisiti

**Gruppo: NightPRO**

[swe.nightpro@gmail.com](mailto:swe.nightpro@gmail.com)

Data: 2025-12-12

Versione: 0.2

## Tabella delle Versioni

Versione	Data	Autore/i	Descrizione delle Modifiche	Verificatore
0,2	2026-01-02	Davide Biasuzzi	Modificato Frontespizio, Inseriti Diagrammi UML degli Use Case, Aggiunto paragrafo sui Requisiti Prestazionali, Aggiornati RF7.4, RF1.1 e RF1.2, Aggiunto paragrafo sui Requisiti di Sicurezza, Completata sezione Stakeholders, Aggiunto UC7.2, Aggiunto Cliente Aziendale sugli utenti principali	Leonardo Bilato
0.1	2025-12-12	Francesco Zanella, Davide Biasuzzi, Samuele Perazzo	Creazione bozza documento con integrazione analisi dettagliata: stakeholder e requisiti funzionali completi (RF1-RF7). Stesura dei casi d'uso (UC1-UC7) strutturati granularmente per garantire l'atomicità delle operazioni e separare le logiche di frontend e backend.	Mihaela-Mariana Romascu

## Indice

<b>Tabella delle Versioni</b>	<b>2</b>
<b>1 Informazioni Generali</b>	<b>5</b>
1.1 Componenti del Gruppo . . . . .	5
<b>2 Introduzione</b>	<b>6</b>
2.1 Scopo del documento . . . . .	6
2.2 Scopo del prodotto . . . . .	6
2.3 Glossario . . . . .	6
2.4 Stakeholder . . . . .	6
<b>3 Dominio d'Uso</b>	<b>7</b>
3.1 Contesto Attuale (AS-IS) . . . . .	7
3.2 Situazione Desiderata (TO-BE) . . . . .	7
<b>4 Descrizione</b>	<b>7</b>
4.1 Obiettivi del prodotto . . . . .	7
4.2 Funzionalità del prodotto . . . . .	8
4.3 Utenti e caratteristiche . . . . .	8
<b>5 Casi d'Uso</b>	<b>9</b>
5.1 Introduzione . . . . .	9
5.2 Attori del Sistema . . . . .	9
5.3 Elenco dei Casi d'Uso . . . . .	10
5.3.1 UC1 - Autenticazione . . . . .	11
5.3.2 UC1.1 - Inserimento Email . . . . .	11
5.3.3 UC1.2 - Inserimento Password . . . . .	12
5.3.4 UC1.3 - Visualizzazione Errore Autenticazione . . . . .	12
5.3.5 UC2 - Visualizzazione Dashboard Ordini . . . . .	12
5.3.6 UC3 - Visualizzazione Dettaglio Ordine . . . . .	13
5.3.7 UC4 - Modifica Riga Ordine . . . . .	13
5.3.8 UC5 - Approvazione e Invio ERP . . . . .	14
5.3.9 UC6 - Inserimento Ordine Testuale . . . . .	15
5.3.10 UC6.1 - Elaborazione Semantica Testo . . . . .	15
5.3.11 UC7 - Inserimento Ordine Vocale . . . . .	16
5.3.12 UC7.1 - Trascrizione Audio (Speech-to-Text) . . . . .	16
5.3.13 UC7.2 - Visualizzazione Errore Trascrizione Audio . . . . .	16
<b>6 Requisiti</b>	<b>17</b>
6.1 Requisiti Funzionali . . . . .	17
6.1.1 Gestione Input Multimodale . . . . .	17
6.1.2 Estrazione e Riconoscimento Entità . . . . .	17
6.1.3 Elaborazione Semantica e Mapping . . . . .	17
6.1.4 Validazione e Arricchimento Dati . . . . .	18
6.1.5 Strutturazione Output e Integrazione Database . . . . .	18
6.1.6 Supervisione e Gestione Manuale . . . . .	18
6.1.7 Monitoraggio e Feedback Continuo . . . . .	19
6.2 Requisiti di Qualità . . . . .	19
6.3 Requisiti di Vincolo . . . . .	19
6.4 Requisiti Prestazionali . . . . .	20

6.5	Requisiti di Sicurezza . . . . .	20
-----	----------------------------------	----

# 1 Informazioni Generali

## 1.1 Componenti del Gruppo

Cognome	Nome	Matricola
Biasuzzi	Davide	2111000
Bilato	Leonardo	2071084
Zanella	Francesco	2116442
Romascu	Mihaela-Mariana	2079726
Ogniben	Michele	2042325
Perozzo	Samuele	2110989
Ponso	Giovanni	2000558

Tabella 1: Componenti del gruppo NightPRO.

## 2 Introduzione

### 2.1 Scopo del documento

Il presente documento descrive in modo chiaro e strutturato i requisiti funzionali e non funzionali del progetto **SmartOrder**. Ha lo scopo di fornire una base di riferimento per tutte le successive fasi del ciclo di vita del software — progettazione, sviluppo, test e validazione — assicurando una visione comune tra tutti i soggetti coinvolti: il team di sviluppo, i docenti, l'azienda proponente e gli utenti finali.

L'obiettivo è definire in modo preciso cosa il sistema dovrà fare, i vincoli tecnici da rispettare e le caratteristiche qualitative che il prodotto dovrà possedere.

### 2.2 Scopo del prodotto

**SmartOrder** è una piattaforma intelligente progettata per automatizzare la gestione degli ordini dei clienti provenienti da diverse tipologie di input: testo e audio. Il sistema utilizza tecniche di **Intelligenza Artificiale (AI)** e **Machine Learning (ML)** per interpretare i dati in modo accurato e trasformarli in ordini strutturati, pronti per essere inseriti automaticamente nel sistema gestionale aziendale (ERP).

**SmartOrder** è in grado di:

- Ricevere ordini tramite email, chat, messaggi vocali;
- Estrarre le informazioni essenziali come articoli, quantità e codici prodotto;
- Validare e normalizzare i dati ottenuti;
- Generare un file JSON pronto per l'importazione nel DB aziendale.

Grazie all'automazione, il sistema riduce l'intervento umano nelle operazioni ripetitive, diminuisce gli errori di interpretazione e aumenta la velocità e l'efficienza dei processi aziendali.

### 2.3 Glossario

Per evitare ambiguità relative alle terminologie utilizzate è stato creato un documento denominato **Glossario**. Questo documento comprende tutti i termini tecnici scelti dai membri del gruppo e utilizzati nei vari documenti con le relative definizioni. Tutti i termini inclusi in questo glossario vengono segnalati all'interno del documento con l'apice **G** accanto alla parola.

### 2.4 Stakeholder

Il progetto **SmartOrder** coinvolge diversi soggetti chiave, ciascuno con ruoli e interessi specifici:

- **Committente:** Docenti Tullio Vardanega e Riccardo Cardin (UNIPD)
- **Product Owner:** Ergon Informatica S.R.L.
- **Consulente aziendale:** Gianluca Carlesso (Ergon Informatica)
- **Utenti finali:** Operatori di gestione ordini e clienti
- **Team di sviluppo:** Membri del gruppo NightPRO, corso di SWE (UNIPD)

## 3 Dominio d'Uso

### 3.1 Contesto Attuale (AS-IS)

Nel panorama odierno, le aziende ricevono richieste di ordini da clienti in forme diverse:

- Testi non strutturati (email, chat, moduli web)
- Messaggi vocali (registrazioni, telefonate trascritte, audio WhatsApp)

La gestione manuale di questi input comporta diverse problematiche:

- Consumo significativo di tempo e risorse umane
- Alto rischio di errori di trascrizione e interpretazione
- Ambiguità nelle descrizioni dei prodotti
- Informazioni incomplete o non standardizzate

### 3.2 Situazione Desiderata (TO-BE)

Il sistema **SmartOrder** mira a creare un ambiente automatizzato capace di:

- Acquisire ordini da molteplici canali e modalità di input
- Interpretare correttamente le intenzioni del cliente nonostante ambiguità e incompletezza
- Validare e normalizzare i dati estratti
- Mappare descrizioni non standardizzate ai codici prodotto aziendali
- Generare ordini strutturati pronti per il database ERP
- Ridurre significativamente l'intervento umano nelle fasi ripetitive

## 4 Descrizione

### 4.1 Obiettivi del prodotto

Gli obiettivi principali di **SmartOrder** sono:

- Automatizzare la ricezione e la gestione degli ordini provenienti da diversi canali;
- Interpretare in modo accurato i contenuti testuali e vocali grazie a modelli di AI;
- Estrarre e strutturare le informazioni necessarie per creare ordini coerenti con il catalogo aziendale;
- Ridurre gli errori legati all'inserimento manuale e alle incomprensioni dei dati;
- Aumentare l'efficienza operativa e migliorare l'esperienza utente.

## 4.2 Funzionalità del prodotto

Il sistema **SmartOrder** è organizzato come una pipeline modulare composta da diverse fasi:

1. **Raccolta degli Input Multimodali:** acquisizione di ordini testuali e vocali con LLM e tecnologie Speech-to-Text.
2. **Pre-Processing dei Dati:** pulizia e normalizzazione degli input con riconoscimento automatico delle entità (NER).
3. **Estrazione delle Caratteristiche:** conversione dei dati multimodali in embedding numerici condivisi.
4. **Interpretazione Semantica e Mappatura:** associazione dei dati ai codici prodotto aziendali.
5. **Validazione e Arricchimento dei Dati:** verifica della completezza e aggiunta di metadati.
6. **Monitoraggio e Feedback:** registrazione dei risultati e miglioramento continuo tramite retraining dei modelli AI.

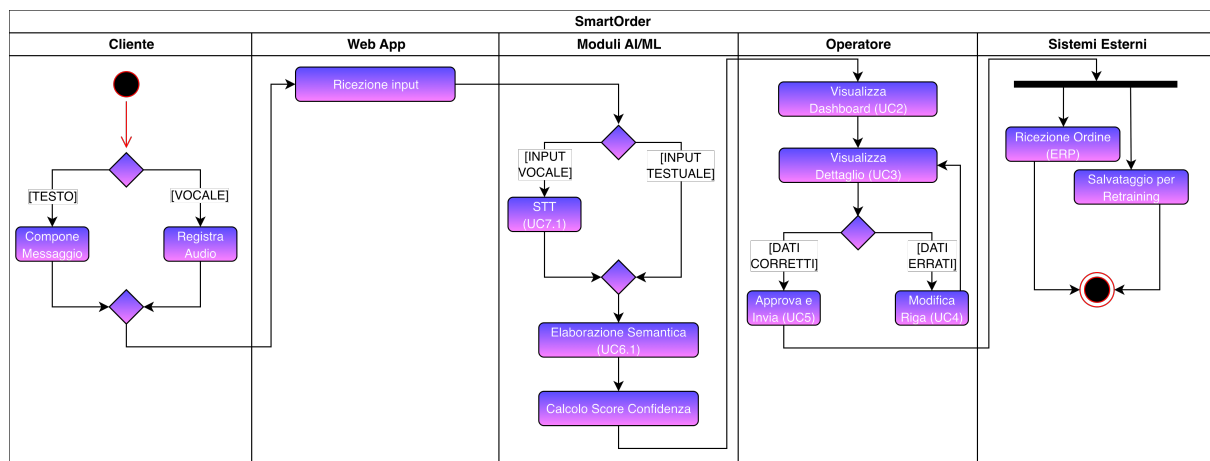


Figura 1: Workflow completo di elaborazione e validazione di un ordine tramite pipeline SmartOrder.

## 4.3 Utenti e caratteristiche

Utenti principali:

Tipo di utente	Descrizione	Obiettivi principali
Operatore aziendale	Supervisiona la creazione e validazione degli ordini generati automaticamente.	Verificare la correttezza degli ordini e risolvere incongruenze.
Cliente Aziendale	Utilizza la webapp per inserire degli ordini tramite input vocale o testuale	Effettuare e revisionare gli ordini.
Amministratore di sistema	Gestisce la base dati e la manutenzione generale del sistema.	Assicurare corretto funzionamento e sicurezza della piattaforma.

Caratteristiche:



- Livello di competenza variabile;
- Interfaccia intuitiva e chiara;
- Accesso tramite web app.

## 5 Casi d'Uso

### 5.1 Introduzione

Questa sezione descrive i casi d'uso del sistema **SmartOrder**. I casi d'uso sono stati ristrutturati per isolare le funzionalità di accesso, le interazioni sulla Web App di gestione e i flussi di acquisizione ordini dai canali esterni.

### 5.2 Attori del Sistema

Il sistema interagisce con i seguenti attori:

- **Cliente:** Utente che invia un ordine tramite uno dei canali supportati (testo, audio).
- **Operatore di gestione:** Utente interno che supervisiona gli ordini processati dal sistema e interviene in caso di ambiguità.
- **Sistema ERP:** Sistema esterno che riceve gli ordini strutturati tramite cartella condivisa.
- **Moduli AI/ML:** Componenti intelligenti del sistema (LLM, speech-to-text).

### 5.3 Elenco dei Casi d'Uso

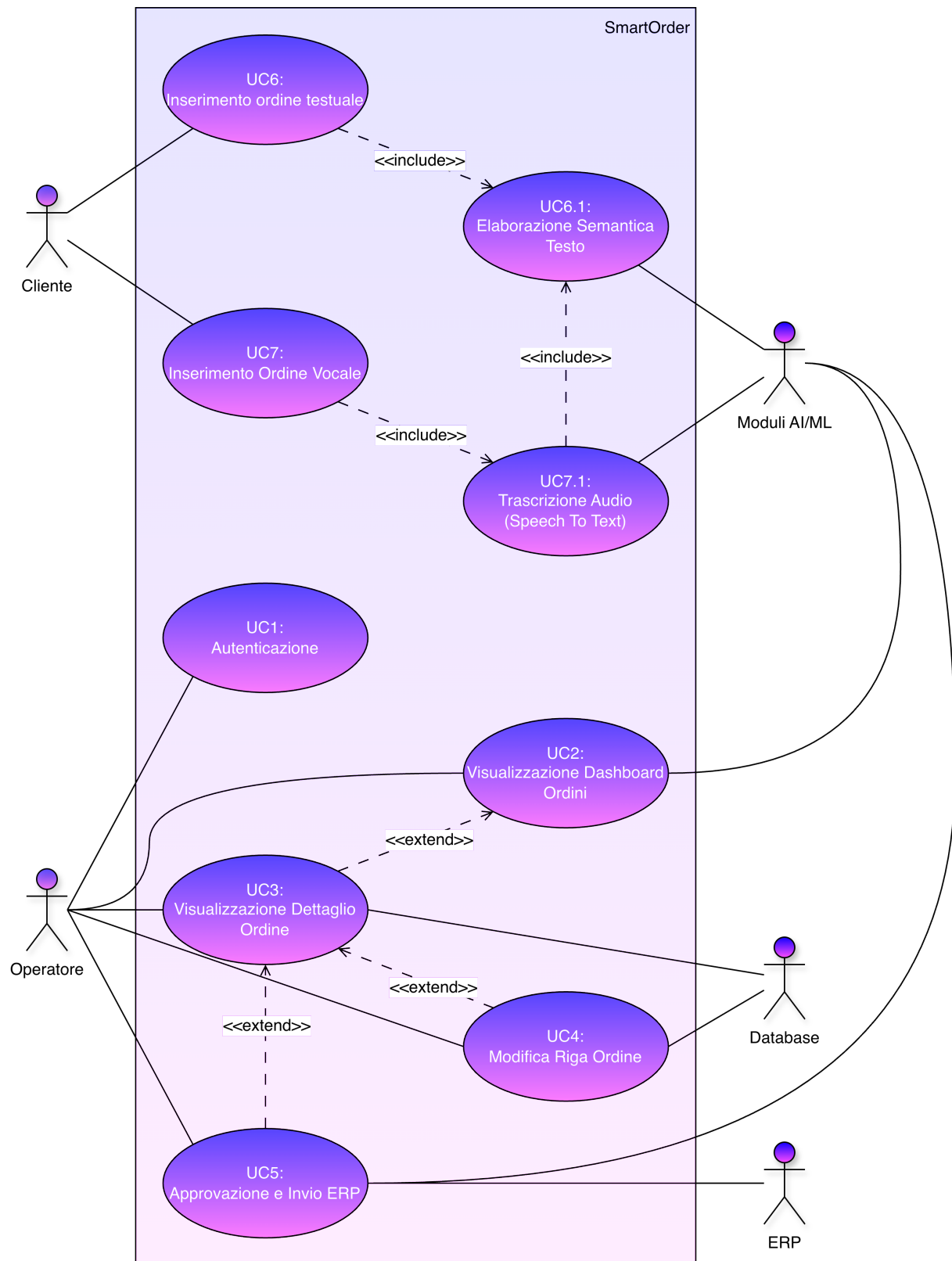


Figura 2: Diagramma Generale dei Casi d'uso

### 5.3.1 UC1 - Autenticazione

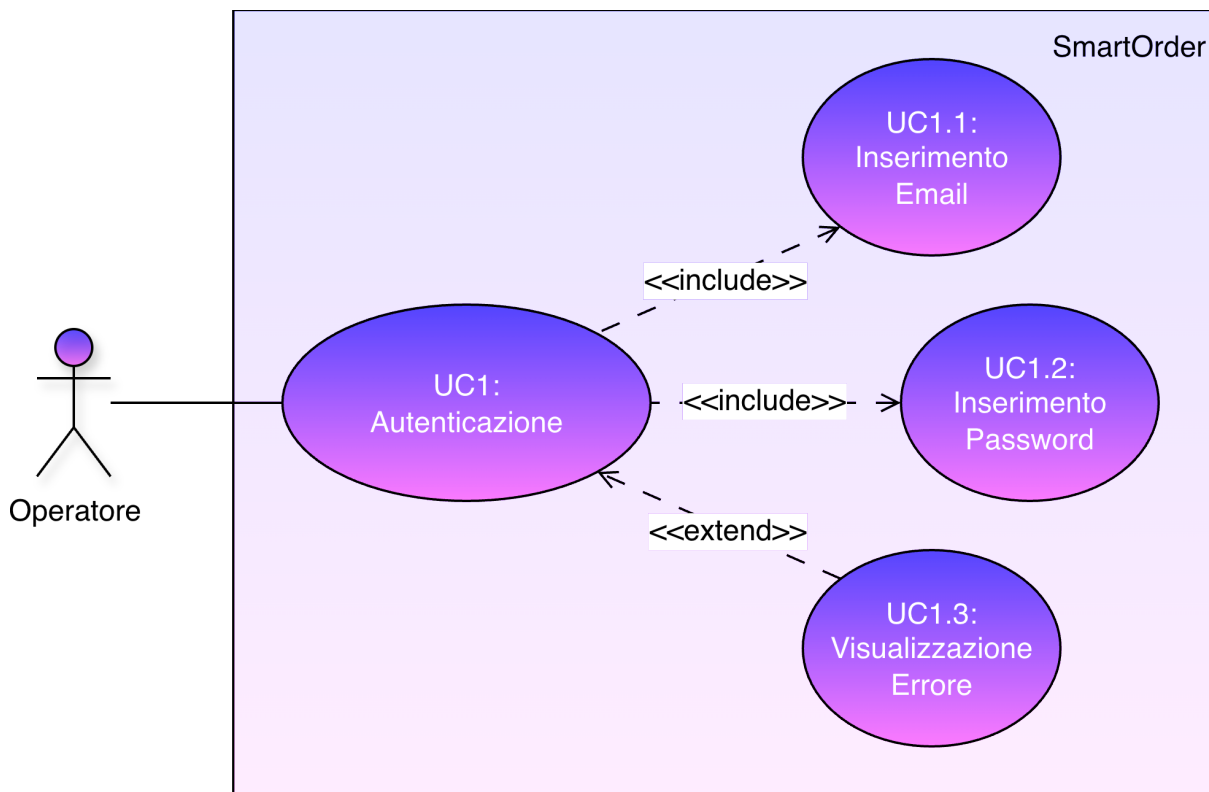


Figura 3: Diagramma UC1

**Attore principale:** Operatore di gestione.

**Precondizioni:** Il sistema è raggiungibile e l'operatore è in possesso delle credenziali.

**Postcondizioni:** L'operatore accede alla dashboard della Web App.

**Scenario principale:**

1. L'operatore visualizza la schermata di login;
2. L'operatore inserisce l'email (vedi UC1.1);
3. L'operatore inserisce la password (vedi UC1.2);
4. Il sistema verifica la validità delle credenziali.

**Estensioni:**

- **Visualizzazione errore (UC1.3):** Se le credenziali sono errate, viene mostrato un messaggio di errore.

**User Story:** Come Operatore, voglio autenticarmi per accedere agli strumenti di gestione ordini.

### 5.3.2 UC1.1 - Inserimento Email

**Attore principale:** Operatore di gestione.

**Precondizioni:** L'operatore sta effettuando l'autenticazione (UC1).

**Postcondizioni:** L'email è inserita nel campo di testo.

**Scenario principale:**

1. L'operatore digita la propria email nel campo dedicato.

### 5.3.3 UC1.2 - Inserimento Password

**Attore principale:** Operatore di gestione.

**Precondizioni:** L'operatore sta effettuando l'autenticazione (UC1).

**Postcondizioni:** La password è inserita nel campo di testo.

**Scenario principale:**

1. L'operatore digita la propria password nel campo dedicato.

### 5.3.4 UC1.3 - Visualizzazione Errore Autenticazione

**Attore principale:** Operatore di gestione.

**Precondizioni:** L'autenticazione (UC1) è fallita a causa di credenziali errate.

**Postcondizioni:** Viene visualizzato un messaggio d'errore.

**Scenario principale:**

1. Il sistema rileva che le credenziali non sono corrette;
2. Il sistema mostra il messaggio "Login fallito. Email o password non validi".

### 5.3.5 UC2 - Visualizzazione Dashboard Ordini

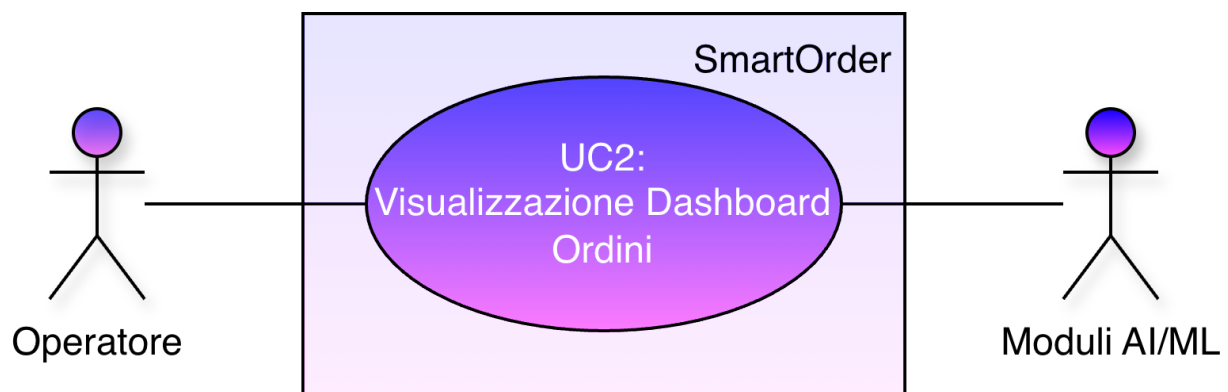


Figura 4: Diagramma UC2

**Attore principale:** Operatore di gestione.

**Precondizioni:** L'operatore ha effettuato l'accesso (UC1).

**Postcondizioni:** L'operatore visualizza la lista degli ordini in attesa di validazione.

**Scenario principale:**

1. L'operatore accede alla sezione principale;
2. Il sistema recupera la lista degli ordini processati dai Moduli AI;
3. Il sistema mostra una tabella riepilogativa (ID, Data, Canale, Stato, Score di confidenza).

**User Story:** Come Operatore, voglio vedere una lista sintetica degli ordini per identificare quali richiedono attenzione.

### 5.3.6 UC3 - Visualizzazione Dettaglio Ordine

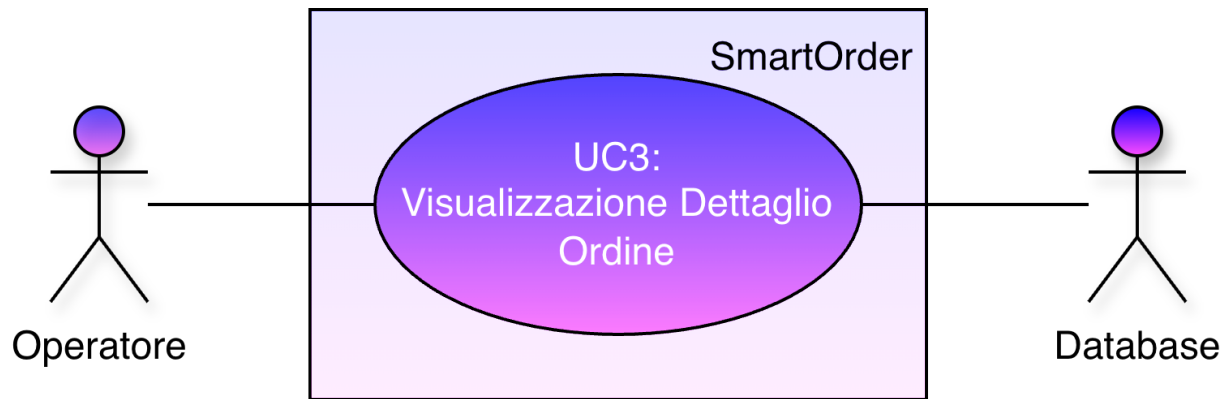


Figura 5: Diagramma UC3

**Attore principale:** Operatore di gestione.

**Precondizioni:** L'operatore ha selezionato un ordine dalla dashboard (UC2).

**Postcondizioni:** Vengono mostrati i dati completi dell'ordine e l'interpretazione AI.

**Scenario principale:**

1. L'operatore seleziona un ordine specifico;
2. Il sistema mostra il contenuto originale (testo o player audio);
3. Il sistema mostra le entità estratte (Prodotti, Quantità) mappate sul catalogo.

**User Story:** Come Operatore, voglio analizzare il dettaglio di un singolo ordine per verificare come l'AI ha interpretato la richiesta.

### 5.3.7 UC4 - Modifica Riga Ordine

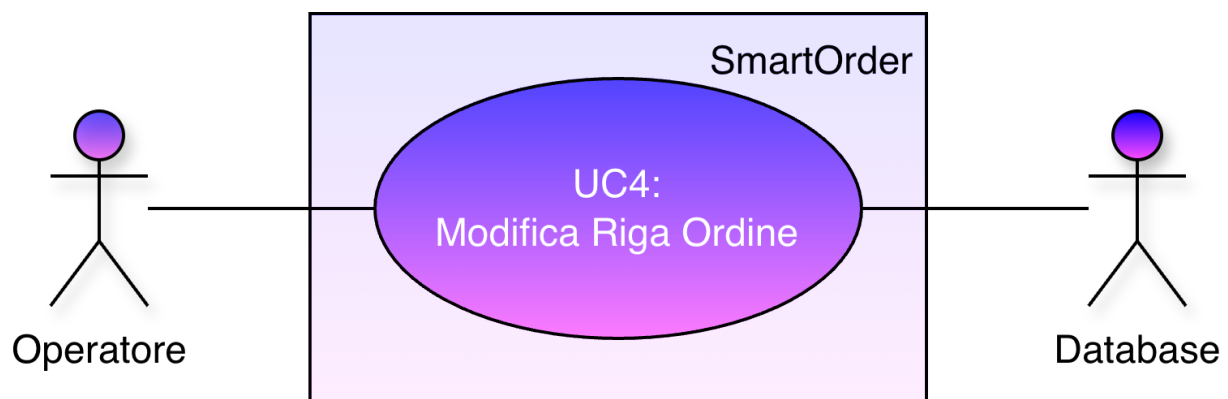


Figura 6: Diagramma UC4

**Attore principale:** Operatore di gestione.

**Precondizioni:** L'operatore si trova nel dettaglio di un ordine (UC3).

**Postcondizioni:** I dati dell'ordine sono aggiornati.

**Scenario principale:**

1. L'operatore identifica un errore di mapping (es. quantità errata o prodotto sbagliato);

2. L'operatore seleziona il campo da modificare;
3. L'operatore inserisce il valore corretto;
4. Il sistema valida e aggiorna temporaneamente l'ordine.

**User Story:** Come Operatore, voglio correggere eventuali errori dell'AI prima di inviare l'ordine al sistema gestionale.

### 5.3.8 UC5 - Approvazione e Invio ERP

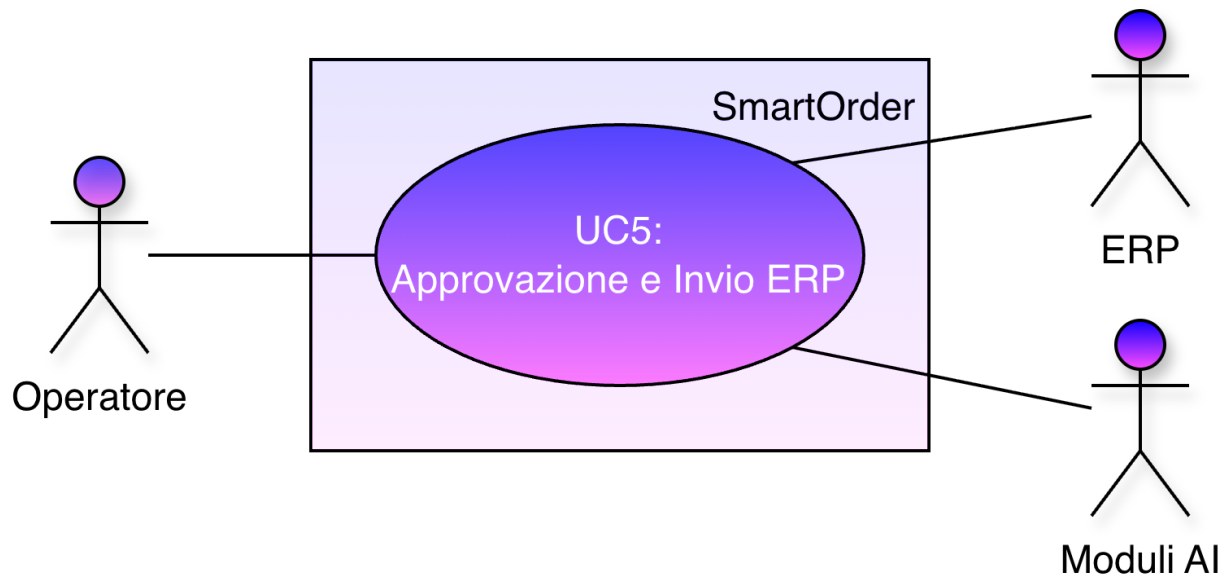


Figura 7: Diagramma UC5

**Attore principale:** Operatore di gestione.

**Precondizioni:** L'ordine è stato verificato ed è pronto.

**Postcondizioni:** L'ordine è trasferito al Sistema ERP e il feedback è salvato.

**Scenario principale:**

1. L'operatore clicca sul comando di approvazione;
2. Il sistema genera il file strutturato (JSON/XML);
3. Il sistema trasferisce il file al **Sistema ERP**;
4. Il sistema salva le correzioni effettuate per il retraining dei **Moduli AI/ML**.

**User Story:** Come Operatore, voglio confermare definitivamente l'ordine affinché venga processato dal magazzino.

### 5.3.9 UC6 - Inserimento Ordine Testuale

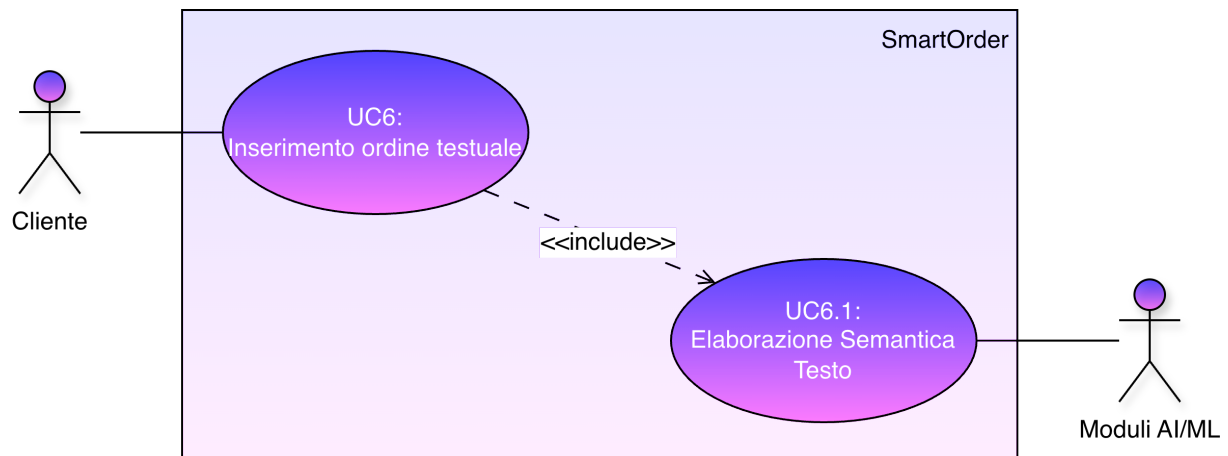


Figura 8: Diagramma UC6

**Attore principale:** Cliente.

**Precondizioni:** Il cliente ha accesso a un canale testuale (es. email, chat).

**Postcondizioni:** Il messaggio è acquisito dal sistema.

**Scenario principale:**

1. Il Cliente compone un messaggio con la lista della spesa;
2. Il Cliente invia il messaggio;
3. Il sistema riceve il testo e avvia l'elaborazione (UC6.1).

**User Story:** Come Cliente, voglio inviare un ordine scrivendo un messaggio naturale senza usare codici complessi.

### 5.3.10 UC6.1 - Elaborazione Semantica Testo

**Attore principale:** Moduli AI/ML.

**Precondizioni:** Un testo è stato ricevuto (UC6) o trascritto (UC7.1).

**Postcondizioni:** Viene generata una bozza strutturata dell'ordine.

**Scenario principale:**

1. Il modulo AI applica algoritmi di NLP/NER sul testo;
2. Vengono estratte le entità (Quantità, Unità, Descrizione);
3. Viene effettuato il mapping con il catalogo prodotti;
4. Il sistema salva la bozza per la visualizzazione operatore (UC2).

### 5.3.11 UC7 - Inserimento Ordine Vocale

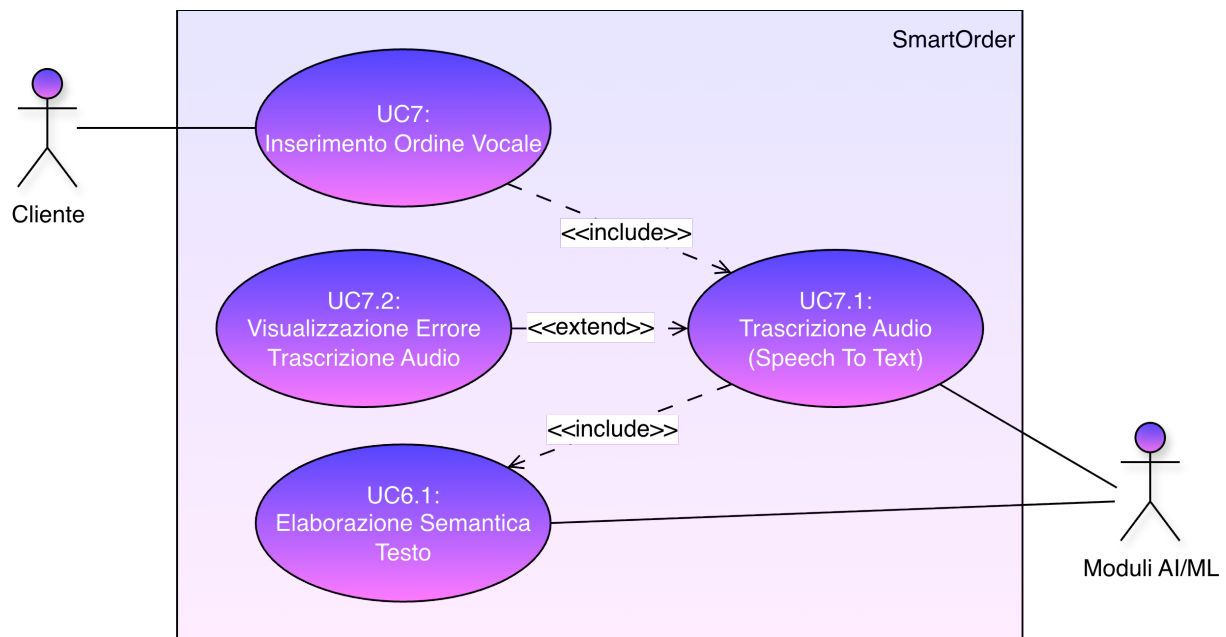


Figura 9: Diagramma UC7

**Attore principale:** Cliente.

**Precondizioni:** Il cliente ha accesso a un canale vocale.

**Postcondizioni:** Il file audio è acquisito dal sistema.

**Scenario principale:**

1. Il Cliente registra un messaggio vocale con l'ordine;
2. Il Cliente invia il file audio;
3. Il sistema riceve il file e avvia la trascrizione (UC7.1).

**User Story:** Come Cliente, voglio dettare il mio ordine vocalmente per risparmiare tempo.

### 5.3.12 UC7.1 - Trascrizione Audio (Speech-to-Text)

**Attore principale:** Moduli AI/ML.

**Precondizioni:** Un file audio è stato ricevuto (UC7).

**Postcondizioni:** Viene prodotto un testo trascritto.

**Scenario principale:**

1. Il modulo AI processa il flusso audio;
2. Il modulo genera una trascrizione testuale;
3. Il sistema passa il testo all'elaborazione semantica (UC6.1).

### 5.3.13 UC7.2 - Visualizzazione Errore Trascrizione Audio

**Attore principale:** Moduli AI/ML.

**Precondizioni:** Un file audio è stato ricevuto (UC7) e la trascrizione è stata tentata (UC7.1).

**Postcondizioni:** Viene segnalato un errore di bassa accuratezza della trascrizione.

**Scenario principale:**



1. Il modulo AI processa il file audio con Speech-to-Text;
2. L'algoritmo determina un confidence score inferiore alla soglia di 90%;
3. Il sistema genera un avviso di errore: "Trascrizione audio non affidabile. Accuratezza: [X]%. Richiesta revisione manuale";
4. L'ordine viene marcato con stato "In revisione - Bassa qualità audio" e segnalato all'operatore in dashboard;
5. L'operatore può visualizzare il file audio originale per effettuare una trascrizione manuale corretta.

**User Story:** Come Sistema, voglio segnalare quando la qualità della trascrizione audio è insufficiente per evitare errori di interpretazione degli ordini.

## 6 Requisiti

### 6.1 Requisiti Funzionali

#### 6.1.1 Gestione Input Multimodale

ID	Requisito	Descrizione	Attori
RF1.1	Acquisizione testo	Il sistema deve acquisire testo dalla chat	Sistema, Cliente
RF1.2	Acquisizione audio	Il sistema deve acquisire file audio (messaggi vocali, registrazioni telefoniche, audio WhatsApp) in MP3 e WAV	Sistema, Cliente
RF1.3	Validazione format input	Il sistema deve validare formato e dimensione degli input	Sistema
RF1.4	Normalizzazione dati grezzi	Il sistema deve normalizzare e ripulire dati grezzi riducendo rumore e ambiguità	Sistema

#### 6.1.2 Estrazione e Riconoscimento Entità

ID	Requisito	Descrizione	Attori
RF2.1	NER (Named Entity Recognition)	Il sistema deve identificare automaticamente entità (quantità, unità di misura, descrizioni prodotto, date di consegna, indirizzi)	Sistema
RF2.2	Speech-to-text	Il sistema deve trascrivere audio in testo con accuracy $\geq 90\%$ (anche per accenti regionali italiani)	Sistema
RF2.3	Tokenizzazione e normalizzazione testuale	Il sistema deve applicare tokenizzazione, lowercasing, rimozione simboli non necessari	Sistema

#### 6.1.3 Elaborazione Semantica e Mapping

ID	Requisito	Descrizione	Attori
RF3.1	Generazione embedding semantici	Il sistema deve generare embedding per testo tramite modelli LLM	Sistema
RF3.2	Generazione embedding audio	Il sistema deve generare embedding dai dati audio trascritti	Sistema
RF3.3	Mapping catalogo prodotti	Il sistema deve mappare descrizioni estratte ai codici prodotto del database aziendale con confidence score	Sistema
RF3.4	Completamento informazioni	Il sistema deve completare o correggere descrizioni incomplete o ambigue basandosi su regole aziendali	Sistema

#### 6.1.4 Validazione e Arricchimento Dati

ID	Requisito	Descrizione	Attori
RF4.1	Verifica integrità ordine	Il sistema deve verificare che ordine contenga informazioni obbligatorie (codice prodotto, quantità, indirizzo)	Sistema
RF4.2	Applicazione regole aziendali	Il sistema deve applicare controlli preliminari e regole aziendali (es. quantità minima, disponibilità magazzino)	Sistema
RF4.3	Validazione coerenza	Il sistema deve validare coerenza tra quantità e unità di misura, codice prodotto e categoria	Sistema
RF4.4	Arricchimento metadati	Il sistema deve arricchire ordine con metadati (canale input, timestamp, tipo input, confidence score)	Sistema
RF4.5	Handling ambiguità	Il sistema deve gestire situazioni ambigue segnalando confidence score basso per validazione manuale	Sistema

#### 6.1.5 Strutturazione Output e Integrazione Database

ID	Requisito	Descrizione	Attori
RF5.1	Strutturazione JSON	Il sistema deve trasformare ordini elaborati in formato JSON strutturato	Sistema
RF5.2	Strutturazione XML	Il sistema deve supportare anche formato XML per compatibilità con sistemi ERP diversi	Sistema
RF5.3	Tracciamento metadati	Il sistema deve tracciare completamente metadati di ogni ordine processato (canale, livello confidenza, timestamp)	Sistema

#### 6.1.6 Supervisione e Gestione Manuale

ID	Requisito	Descrizione	Attori
RF6.1	Interfaccia supervisione	Il sistema deve fornire interfaccia web per operatore di gestione per supervisionare ordini	Operatore, Sistema
RF6.2	Visualizzazione dettagli ordine	L'interfaccia deve mostrare dettagli completi dell'ordine (dati estratti, confidence score, sorgente input)	Operatore
RF6.3	Correzione manuale	L'interfaccia deve permettere operatore di correggere dati ordine prima dell'inserimento in caso di ambiguità	Operatore, Sistema
RF6.4	Approvazione/Rifiuto	L'interfaccia deve permettere operatore di approvare o rifiutare ordine ambiguo	Operatore, Sistema
RF6.5	Feedback per retraining	Il sistema deve registrare correzioni manuali per retraining del modello	Sistema
RF6.6	Visualizzazione storico	L'interfaccia deve permettere accesso a storico degli ordini processati con filtri (data, canale, cliente)	Operatore

### 6.1.7 Monitoraggio e Feedback Continuo

ID	Requisito	Descrizione	Attori
RF7.1	Logging dettagliato	Il sistema deve loggare tutte le operazioni con timestamp, input, output, errori	Sistema
RF7.2	Monitoraggio prestazioni	Il sistema deve tracciare metriche di performance (tempo processing, accuracy, error rate)	Sistema
RF7.3	Meccanismo feedback	Il sistema deve implementare meccanismo di feedback continuo per correzione dei dati	Sistema, Operatore
RF7.4	Retraining periodico	Il sistema deve supportare retraining periodico (periodicità a scelta) e manuale dei modelli AI basato su dati corretti	Sistema
RF7.5	Aggiornamento regole	Il sistema deve permettere aggiornamento delle regole aziendali in base a pattern osservati	Sistema

## 6.2 Requisiti di Qualità

Codice	Classificazione	Descrizione	Fonte
RQ01	Desiderabile	Il sistema deve fornire messaggi di errore chiari e comprensibili in caso di input non valido.	Team di sviluppo
RQ02	Desiderabile	Il sistema deve poter essere esteso per supportare nuovi canali di input (es. video, moduli web).	Capitolato

## 6.3 Requisiti di Vincolo

Codice	Classificazione	Descrizione	Fonte
RV01	Obbligatorio	Il database deve essere relazionale (MySQL, MariaDB o SQL Server Express).	Capitolato

RV02	Obbligatorio	Le API devono rispettare lo standard REST per garantire interoperabilità.	Capitolato
RV03	Desiderabile	L'interfaccia deve essere sviluppata come webapp responsive.	Capitolato
RV04	Obbligatorio	Il sistema deve essere compatibile con i principali browser web moderni.	Team di sviluppo
RV05	Desiderabile	L'addestramento dei modelli deve poter essere eseguito in ambiente locale o cloud.	Capitolato

## 6.4 Requisiti Prestazionali

In quanto soluzione Web-based, i requisiti prestazionali sono strettamente correlati alla latenza e alla larghezza di banda della connessione Internet, nonché alle capacità computazionali del terminale utente. L'ottimizzazione del frontend è stata declinata specificamente per motori di rendering Chromium, garantendo piena compatibilità con browser quali Google Chrome e Microsoft Edge.

## 6.5 Requisiti di Sicurezza

Il progetto definisce i meccanismi di sicurezza a livello applicativo, quali l'autenticazione degli utenti e la gestione dei permessi basata sui ruoli (RBAC). Tuttavia, la messa in sicurezza dell'infrastruttura di produzione (hardening dei server, configurazione di firewall, gestione dei certificati SSL/TLS e conformità GDPR) esula dagli obiettivi di questo elaborato. Tali aspetti saranno presi in carico dall'ente responsabile dell'implementazione finale e del deployment nell'ambiente operativo aziendale.