

## Gruppo OOB2324\_18

Davide Gargiulo	david.gargiulo@studenti.unina.it	N86004689
Vincenzo Di Carluccio	vi.dicarluccio@studenti.unina.it	N86004800

# UNINA DELIVERY



UNIVERSITÀ  
DEGLI STUDI  
DI NAPOLI  
FEDERICO II



# Contents

<b>1</b>	<b>Introduzione</b>	<b>3</b>
1.1	Scopo Del Documento . . . . .	3
1.2	Descrizione Del Sistema . . . . .	3
1.3	Utenti Intesi/Destinatari . . . . .	3
<b>2</b>	<b>Progettazione concettuale</b>	<b>4</b>
2.1	Diagramma Delle Classi UML . . . . .	4
2.2	Leggenda del Diagramma Entità-Relazione . . . . .	5
2.3	Diagramma Entità-Relazione . . . . .	6
<b>3</b>	<b>Ristrutturazione</b>	<b>7</b>
3.1	Considerazioni . . . . .	7
3.1.1	Attributi Multivalore e Composti . . . . .	7
3.1.2	Specializzazioni . . . . .	7
3.1.3	Attributi Derivati . . . . .	7
3.1.4	Analisi delle Ridondanze . . . . .	8
3.1.5	Identificazione delle Chiavi Primarie . . . . .	8

# 1 Introduzione

## 1.1 Scopo Del Documento

Questo documento fornisce una panoramica dettagliata di UninaDelivery, un sistema per la gestione della logistica e delle spedizioni di merci. Il documento è stato concepito per offrire una guida completa agli utenti del sistema, delineando le sue funzionalità, architettura e il flusso di lavoro associato.

## 1.2 Descrizione Del Sistema

Il database di UninaDelivery è stato progettato per ottimizzare la gestione logistica delle spedizioni di merci basandosi sugli ordini dei clienti. Il sistema permette agli operatori di pianificare in modo efficiente le spedizioni, tenendo conto di variabili cruciali come la disponibilità della merce, il suo peso, e la disponibilità di mezzi di trasporto e corrieri. Questa soluzione tecnologica mira a migliorare l'efficienza operativa, ridurre i tempi di consegna e massimizzare la soddisfazione del cliente.

## 1.3 Utenti Intesi/Destinatari

Questa documentazione è stata creata pensando in particolare ai seguenti destinatari:

- **Operatori Logistici:** che utilizzeranno il sistema quotidianamente per la gestione e pianificazione delle spedizioni;
- **Team di Supporto e Manutenzione:** responsabili della manutenzione e dell'aggiornamento del sistema;
- **Dirigenti Aziendali:** che necessitano di comprendere le capacità e i benefici del sistema per prendere decisioni informate a livello strategico;
- **Sviluppatori di Software:** che potrebbero avere bisogno di interfacciarsi con UninaDelivery per integrazioni o personalizzazioni.

## 2 Progettazione concettuale

### 2.1 Diagramma Delle Classi UML

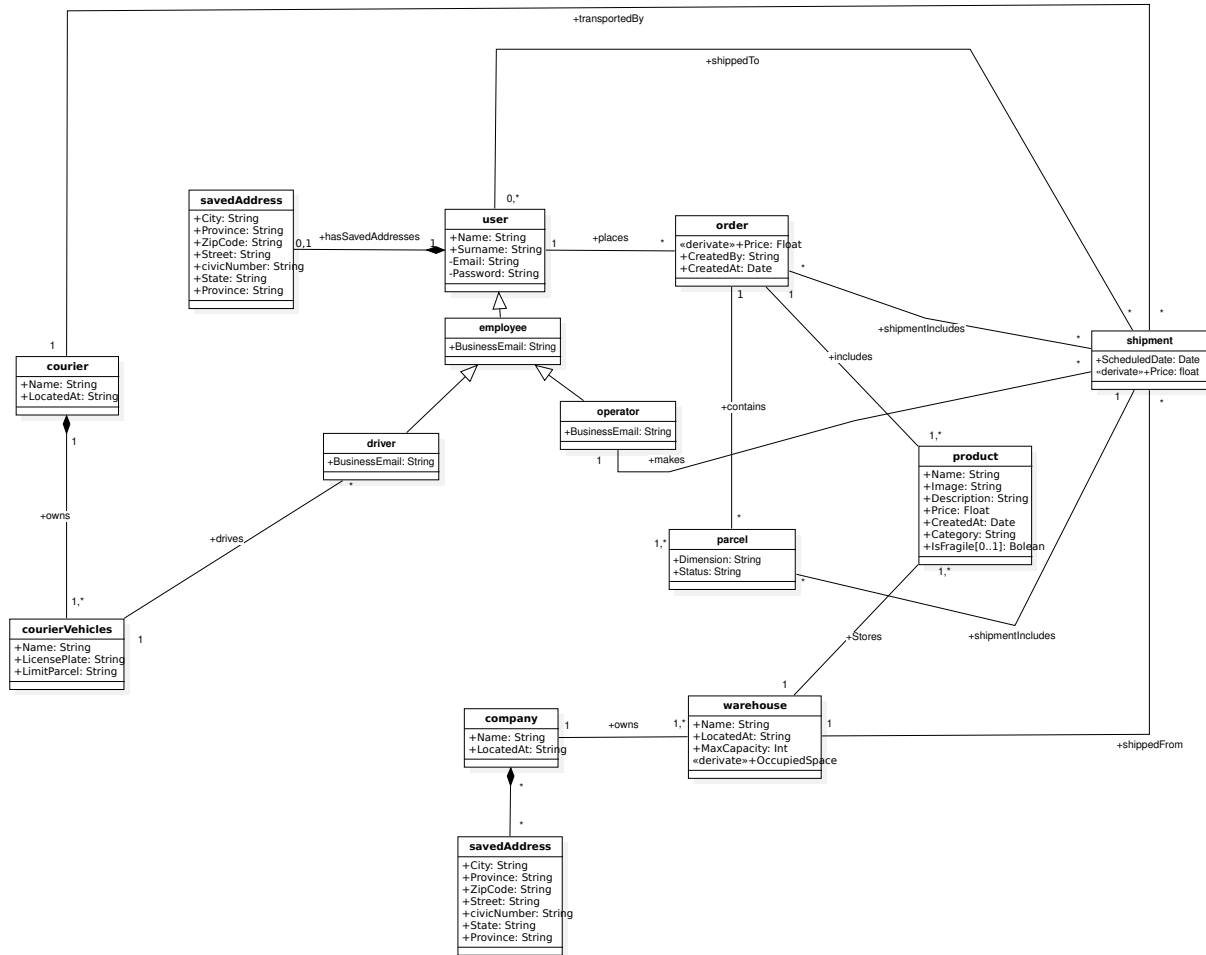


Figure 1: Diagramma delle classi UML

## 2.2 Leggenda del Diagramma Entità-Relazione

Nella creazione del diagramma Entità-Relazione, abbiamo impiegato diversi colori per distinguere chiaramente le varie componenti. Per assicurare la chiarezza e facilitare la comprensione del diagramma, di seguito è presentata una leggenda che spiega i simboli usati e i loro colori corrispondenti:

- **Entità:**
  - **Rettangolo Azzurro:** Rappresenta un'entità. Il nome dell'entità è posizionato all'interno del rettangolo;
  - **Rettangolo Grigio con doppia linea:** Indica un'entità debole.
- **Attributo:**
  - **Ovale Giallo:** Rappresenta un attributo. Il nome dell'attributo è posizionato all'interno dell'ovale;
  - **Ovale tratteggiato:** Indica un attributo derivato;
  - **Nome sottolineato:** Indica un attributo chiave.
- **Relazione:**
  - **Rombo Verde:** Rappresenta una relazione. Il nome della relazione è posizionato all'interno del rombo;
  - **Rombo con Doppia Linea:** Indica una relazione di tipo debole.
- **Specializzazioni:**
  - **Cerchio e Linea Rosa:** Rappresentano una specializzazione.
    - **Lettera "U" (Magnetite):** Indica la direzione della sottoclasse nella specializzazione;
    - **Lettera "d":** Specifica una specializzazione disgiunta;
    - **Lettera "o":** Specifica una specializzazione overlapping;
    - **Singola Linea:** Rappresenta la parzialità della specializzazione;
    - **Doppia Linea:** Rappresenta la totalità della specializzazione.

## 2.3 Diagramma Entità-Relazione

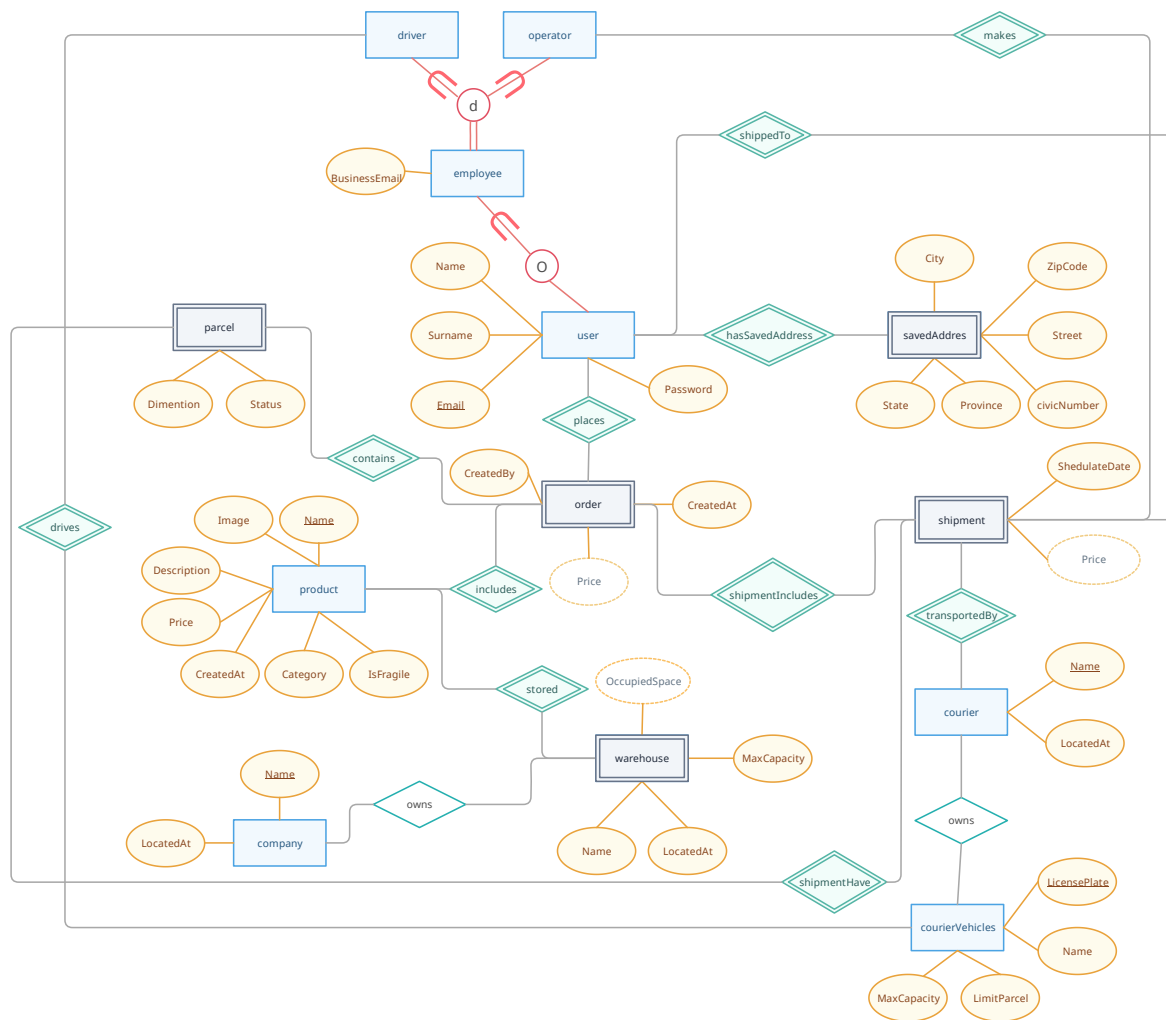


Figure 2: Diagramma Entità-Relazione

## 3 Ristrutturazione

### 3.1 Considerazioni

#### 3.1.1 Attributi Multivalore e Composti

Durante la fase di progettazione concettuale del database, è stata presa una decisione riguardo agli attributi multivalore e composti. Abbiamo scelto di evitare l'uso di attributi multivalore o composti. Questa decisione è stata guidata dalla necessità di mantenere una struttura di database chiara e di facile gestione.

#### 3.1.2 Specializzazioni

Le varie **specializzazioni** e le varie **generalizzazioni** sono state modellate nei seguenti modi:

- **Employee:**
  - **Tipo di Specializzazione:** Totale e disgiunta;
  - **Implementazione:** Abbiamo scelto di accorpare la classe generale **Employee** in classi specializzate.
- **User:**
  - **Tipo di Specializzazione:** Parziale e Totale;
  - **Implementazione:** Per la specializzazione di **Account**, abbiamo deciso di trasformarla in un'associazione. Questo approccio ci ha permesso di limitare i vincoli d'integrità e di gestire in modo più efficace i campi che possono assumere valori null.

#### 3.1.3 Attributi Derivati

Nel contesto della progettazione concettuale del database, abbiamo implementato tre attributi derivati, gestendoli nel seguente modo:

- **Attributo Price:**
  - **Gestione:** Nonostante **Price** sia un attributo derivato (tipicamente calcolato a partire da altri dati), abbiamo deciso di conservarlo direttamente nel nostro sistema.
  - **Motivazione:** Questa scelta è stata fatta per ottimizzare le prestazioni del sistema, evitando il calcolo ripetuto dei prezzi ogni volta che vengono richiesti. Conservando il prezzo calcolato, riduciamo il carico sul sistema e miglioriamo l'efficienza delle query.
- **Attributo OccupiedSpace:**
  - **Gestione:** Abbiamo scelto di conservare anche l'attributo **OccupiedSpace** all'interno del nostro sistema.
  - **Motivazione:** Questa decisione è stata presa considerando l'importanza di questo attributo, che può essere frequentemente richiesto dal sistema per la gestione degli spazi nei magazzini. Conservandolo, siamo in grado di fornire rapidamente le informazioni richieste senza la necessità di calcoli aggiuntivi.

### 3.1.4 Analisi delle Ridondanze

Nel processo di ottimizzazione del database, abbiamo eseguito un'analisi dettagliata per identificare e risolvere eventuali ridondanze.

- **Ridondanza negli attributi delle associazioni:**

- **Situazione Identificata:** Dopo la ristrutturazione delle specializzazioni, abbiamo riscontrato la presenza di ridondanza negli attributi. In particolare, l'attributo **BusinessEmail** era presente sia nell'associazione **Driver** che in **Operator**, pur essendo associato a due entità diverse.
- **Soluzione Implementata:** Per risolvere questa ridondanza, abbiamo introdotto attributi identificativi specifici per ciascuna associazione. Nell'associazione **Driver**, è stato aggiunto l'attributo **driverId** per identificare univocamente ogni driver. Analogamente, nell'associazione **Operator**, è stato introdotto **operatorId** per l'identificazione univoca degli operatori.

- **Ambiguità nei nomi delle relazioni:**

- **Situazione Identificata:** Abbiamo notato una sovrapposizione nei nomi delle relazioni tra **Order** e **Shipment** e tra **Parcel** e **Shipment**. Nonostante avessero lo stesso nome, queste relazioni indicavano relazioni sostanzialmente diverse.
- **Soluzione Implementata:** Per eliminare questa ambiguità, abbiamo deciso di rinominare la relazione tra **Order** e **Shipment** in **orderShipment**. Questo aiuta a distinguere chiaramente questa relazione da quella tra **Parcel** e **Shipment**.

### 3.1.5 Identificazione delle Chiavi Primarie

Abbiamo effettuato delle scelte riguardo all'uso di chiavi primarie, in particolare l'introduzione di chiavi surrogate in alcune associazioni chiave.

- **Associazione Order:**

- **Decisione:** Introduzione di una chiave surrogate per identificare univocamente ogni ordine chiamata **OrderId**.
- **Motivazione:** L'uso di una chiave surrogate in questa associazione assicura che ogni ordine registrato nel sistema sia univoco. Questo non solo semplifica la gestione e il tracciamento degli ordini attuali, ma apre anche la strada per un potenziale supporto di funzionalità avanzate, come la gestione di ordini multipli o complessi in futuro.

- **Associazioni Driver e Operator:**

- **Decisione:** Aggiunta di chiavi surrogate in entrambe le associazioni, come menzionato nell'analisi delle ridondanze.
- **Motivazione:** Questa scelta è stata guidata dall'analisi della ridondanza, garantendo un'identificazione chiara e distinta di ogni driver e ogni operatore.

- **Associazione Company:**

- **Decisione:** Introduzione di una chiave surrogate nell'associazione **Company**, chiamata **CompanyId**.
- **Motivazione:** Abbiamo identificato la necessità di un identificatore unico più affidabile del solo nome aziendale, avendo riscontrato casi di aziende con lo stesso nome. La chiave surrogate assicura una distinzione precisa tra le aziende, anche in presenza di nomi omonimi.