



# UninaFoodLab

Antonino De Martino  
N86005103

Simone Barbella  
N86004906

Anno Accademico 2024/2025

# Indice

<b>1</b>	<b>Presentazione del Progetto</b>	<b>3</b>
1.1	Cos'è UninaFoodLab . . . . .	3
<b>2</b>	<b>Progettazione Concettuale</b>	<b>4</b>
2.1	Diagramma ER . . . . .	4
2.1.1	Descrizione Diagramma ER . . . . .	5
2.1.2	Gestione Utenti e Gerarchie . . . . .	5
2.1.3	Corsi e Iscrizioni . . . . .	5
2.1.4	Struttura delle Sessioni . . . . .	5
2.1.5	Logistica in Presenza (Adesioni e Ricette) . . . . .	6
2.2	Diagramma UML . . . . .	7
2.2.1	Descrizione Diagramma UML . . . . .	7
2.2.2	Tipi di Dato e Precisione Numerica . . . . .	7
2.2.3	Rappresentazione degli Array (Attributi Multivalore) . . . . .	8
2.3	Ristrutturazione Diagramma UML . . . . .	9
2.3.1	Descrizione Delle Scelte . . . . .	9
2.3.2	Accorpamento della gerarchia Utente . . . . .	10
2.3.3	Accorpamento della gerarchia Sessione . . . . .	10
<b>3</b>	<b>Dizionari</b>	<b>11</b>
3.1	Dizionario delle Classi . . . . .	11
3.2	Dizionario Associazioni . . . . .	13
3.3	Dizionario Vincoli . . . . .	14
<b>4</b>	<b>Schema Logico</b>	<b>18</b>
4.1	Tabelle . . . . .	18

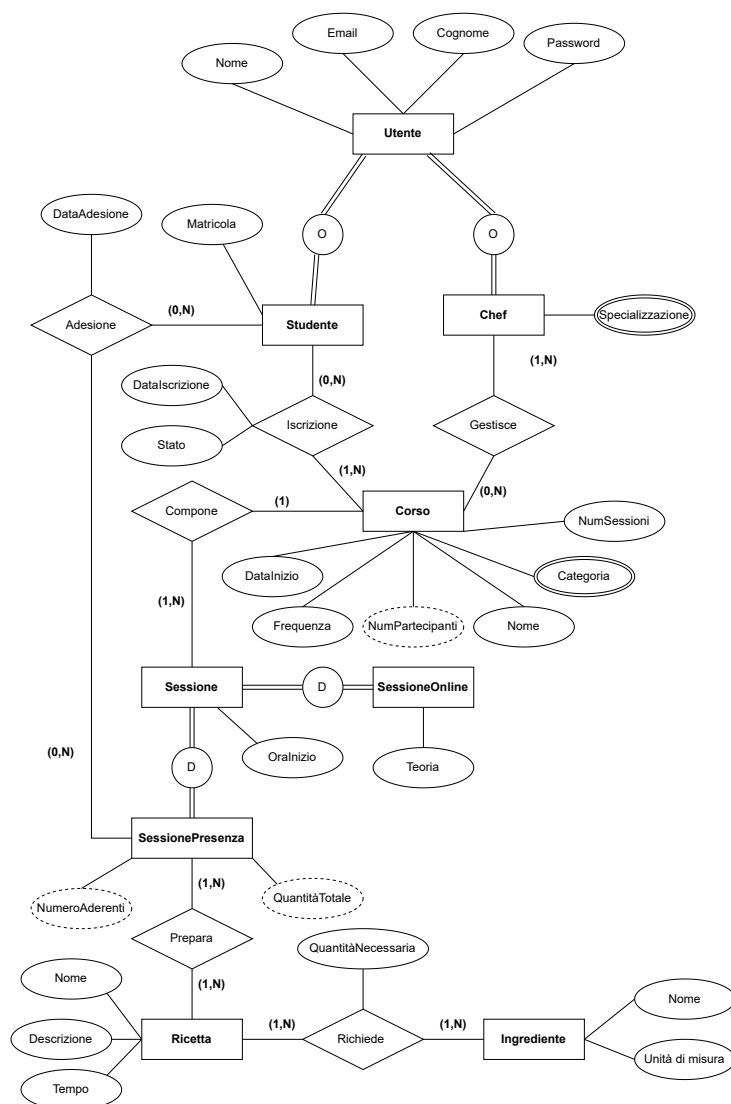
# 1 Presentazione del Progetto

## 1.1 Cos'è UninaFoodLab

UninaFoodLab nasce per modernizzare l'insegnamento culinario, unendo la flessibilità dell'apprendimento digitale con l'imprescindibile esperienza pratica in laboratorio. L'obiettivo principale non è solo la gestione didattica, ma l'ottimizzazione delle risorse: il sistema è progettato per calcolare con precisione i fabbisogni di materie prime, riducendo drasticamente lo spreco alimentare (Food Waste) attraverso un meccanismo di conferme puntuali.

## 2 Progettazione Concettuale

### 2.1 Diagramma ER



### 2.1.1 Descrizione Diagramma ER

Il diagramma descrive il database per la piattaforma UninFoodLab, includendo la gestione degli utenti, la pianificazione delle sessioni e la logistica degli ingredienti.

### 2.1.2 Gestione Utenti e Gerarchie

L'entità padre è Utente, che raccoglie i dati comuni: *Nome*, *Cognome*, *Email* e *Password*. Esiste una generalizzazione che divide l'utente in due ruoli. La doppia linea sotto "Utente" suggerisce una generalizzazione totale, mentre i cerchi con la lettera "O" indicano una specializzazione overlapping:

- **Studente**: Identificato da una Matricola;
- **Chef**: Caratterizzato da una Specializzazione.

### 2.1.3 Corsi e Iscrizioni

L'entità centrale dell'offerta formativa è il Corso.

- Relazione **Gestisce**: Collega Chef e Corso.
  - Uno Chef gestisce da 0 a N corsi;
  - Un corso deve essere gestito da almeno 1 chef per esistere (cardinalità 1,N)f.
- Relazione **Iscrizione**: Collega Studente e Corso.
  - È una relazione multi-a-molti con attributi propri: *DataIscrizione* e *Stato*;
  - Uno studente può avere 0 a N iscrizioni;
  - Un corso deve avere almeno 1 iscritto per esistere (cardinalità 1,N).

### 2.1.4 Struttura delle Sessioni

Il corso è strutturato in lezioni tramite la relazione **Compone**.

- Cardianalità: Un Corso è composto da 1 a N sessioni , mentre una Sessione appartiene a 1 solo corso;
- Entità Sessione : Ha un attributo OraInizio;
- Gerarchia Sessioni: È presente una generalizzazione Disgiunta (indicata dalla lettera "D" ), il che significa che una sessione è o online o in presenza, mai entrambe;

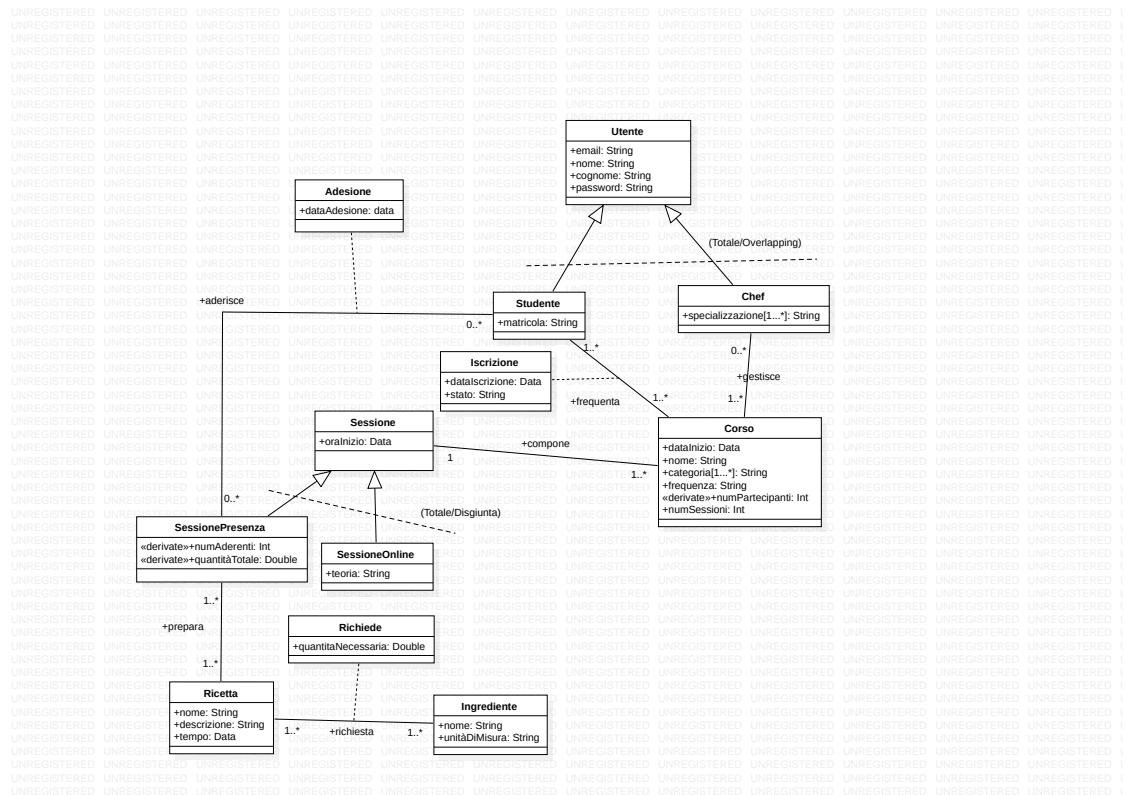
- SessioneOnline: Ha l'attributo specifico *Teoria*.
- SessionePresenza: È l'entità che gestisce la logistica fisica.

### 2.1.5 Logistica in Presenza (Adesioni e Ricette)

Questa parte è specifica per le SessionePresenza:

- Adesione Studenti: Gli studenti partecipano fisicamente tramite la relazione **Adesione**, che ha l'attributo *DataAdesione*;
  - La SessionePresenza ha un attributo derivato *NumeroAderenti* per contare i presenti;
- Preparazione Ricette: La relazione **Prepara** collega la sessione alle ricette;
  - Ha un attributo derivato importante: *QuantitàTotale*;
  - Cardinalità: Una sessione prepara 1 a N ricette , e una ricetta può essere preparata in 1 a N sessioni.

## 2.2 Diagramma UML



### 2.2.1 Descrizione Diagramma UML

### 2.2.2 Tipi di Dato e Precisione Numerica

Il diagramma UML definisce esattamente come i dati devono essere memorizzati, sciogliendo le ambiguità generiche dell'ER:

- Gestione delle Quantità: Si nota una distinzione importante tra Int e Double;
- Contatori come *numPartecipanti* e *numAderenti* sono Int, poiché contano unità discrete (persone, lezioni);
- Le misure fisiche come **quantitàTotale** e **quantitàNecessaria** sono definite come Double. Questo indica che il sistema gestirà valori decimali precisi (es. 1.5 kg, 0.25 litri), dettaglio fondamentale per le ricette che l'ER lasciava implicito.

- Date e Orari: i riferimenti temporali sono distinti in base al significato: *dataAdesione* e *dataIscrizione* sono Date, *oraInizio* è un DateTime (data+ora), mentre *tempo* è un Time (ora, minuti, secondi).

### 2.2.3 Rappresentazione degli Array (Attributi Multivalore)

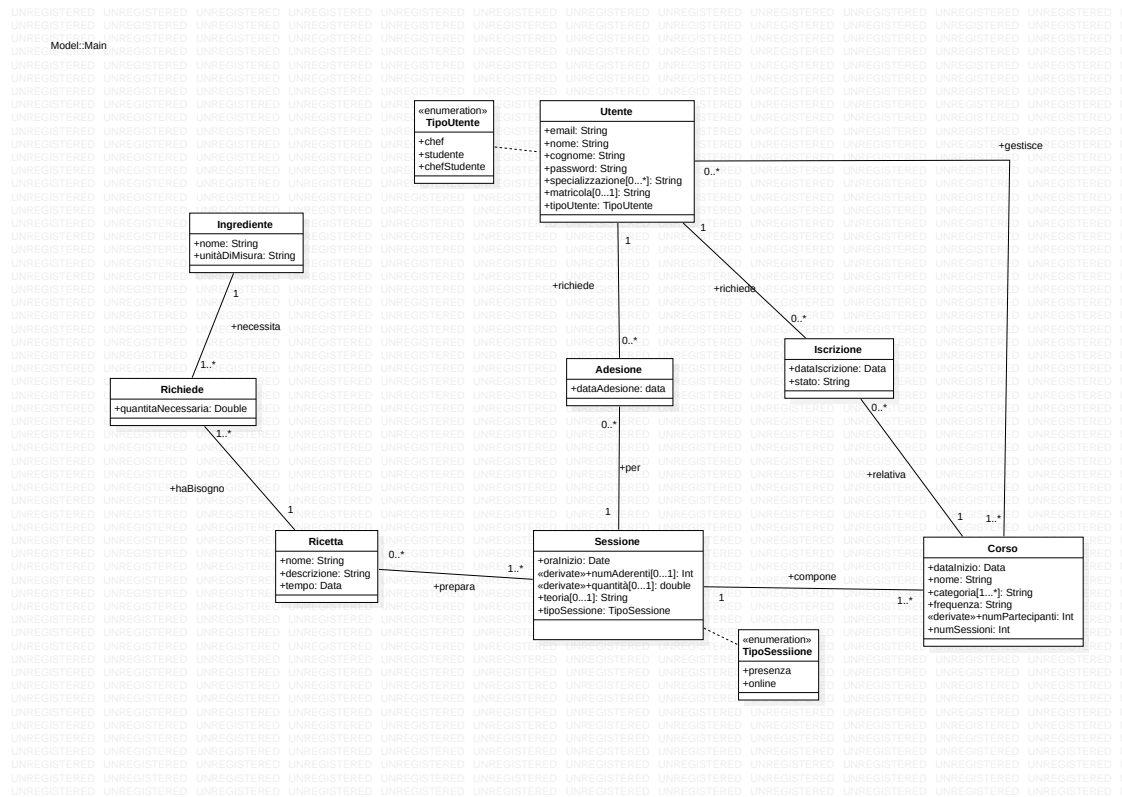
L'UML traduce graficamente gli attributi multivalore dell'ER con la notazione delle parentesi quadre:

- +specializzazione[1...\*]: String nella classe Chef;
- +categoria[1...\*]: String nella classe Corso.

Questa notazione dice esplicitamente che posso avere più valori.



## 2.3 Ristrutturazione Diagramma UML



### 2.3.1 Descrizione Delle Scelte

Nel modello iniziale era presente una generalizzazione della classe Utente nelle sottoclassi Studente e Chef. Abbiamo deciso di adottare la seconda strategia, accorpando le due sottoclassi all'interno della classe padre Utente. Questa scelta è motivata dall'assenza di attributi propri nella classe Studente, che avrebbe reso la distinzione strutturale poco significativa. Per mantenere comunque la distinzione dei ruoli, è stato introdotto un attributo discriminante *TipoUtente*.

Analogamente, nel modello iniziale era presente una generalizzazione della classe Sessione nelle sottoclassi SessionePresenza e SessioneOnline. Anche in questo caso è stata adottata la strategia di accorpamento nella classe padre, in quanto la classe SessioneOnline offriva un contributo informativo limitato e non tale da giustificare una modellazione separata. È stato pertanto introdotto un attributo discriminante *TipoSessione*. Questo attributo consente di gestire correttamente la natura disgiunta della generalizzazione e di distinguere in modo chiaro le diverse modalità di erogazione delle sessioni.

### 2.3.2 Accorpamento della gerarchia Utente

Invece di avere tre classi distinte (Utente, Chef, Studente), ora esiste una sola classe Utente:

- Enumerazione come discriminatore: È stata introdotta l'enumerazione `TipoUtente` che contiene i valori **chef**, **studente** e il valore combinato **chefStudente**. Questo gestisce esplicitamente il caso di Overlapping;
- Fusione degli attributi: La classe Utente contiene ora tutti gli attributi possibili di tutti i ruoli, ma con cardinalità facoltativa:
  - `+matricola[0...1]`: String: Sarà valorizzata solo se l'utente è uno studente.
  - `+specializzazione[0...*]`: String: Sarà valorizzata solo se l'utente è uno chef.

### 2.3.3 Accorpamento della gerarchia Sessione

Analogamente a Utente, la distinzione tra sessioni online e in presenza è stata rimossa strutturalmente per creare un'unica classe Sessione:

- Enumerazione come discriminatore: È introdotta l'enumerazione `TipoSessione` che contiene i valori *Presenza* e *Online* per distinguere la natura della lezione.
- Fusione degli attributi: La classe Sessione contiene ora tutti gli attributi che prima erano separati, rendendoli però opzionali:
  - `+teoria[0...1]`: String: Usato solo se il tipo è online;
  - `«derivate»numAderenti[0...1]`: Int e `«derivate»+quantità[0...1]`: double: Usati solo se il tipo è presenza.

## 3 Dizionari

### 3.1 Dizionario delle Classi

Table 1: Dizionario delle Classi:1

Classe	Descrizione	Attributi
Utente	Descrittore di ogni Studente o Chef	<b>email</b> (String): è un indirizzo email unico e valido che permette di contattare l'utente. <b>nome</b> (String): è il nome dell'utente. <b>cognome</b> (String): è cognome dell'utente <b>password</b> (String): una password permette all'utente di proteggere il suo account e le sue informazioni. <b>specializzazione</b> (String)[0..*]: insieme di specializzazioni dello chef (realizzato tramite tabella <b>Specializzazione_Chef</b> ). <b>matricola</b> (String): string alfanumerica che identifica lo studente <b>tipoUtente</b> (TipoUtente): indica il tipo di utente: (chef), (studente) oppure (chefStudente)
Corso	Descrittore di ciascun corso creato.	<b>dataInizio</b> (Data): indica l'inizio del corso. <b>nome</b> (String): è il nome assegnato al corso. <b>categoria</b> (String)[1..*]: insieme di categorie del corso (realizzato tramite tabella <b>Categoria_Corso</b> ). <b>frequenza</b> (String): indica ogni quanti giorni il corso si svolge <b>numPartecipanti</b> (Int): Numero dei partecipanti al corso <b>numSessioni</b> (Int): specifica quante sessioni ha un corso.
Iscrizione	Descrittore dell'iscrizione degli utenti.	<b>dataIscrizione</b> (Data): indica l'inizio dell'iscrizione per un corso. <b>Stato</b> (String): indica lo stato dell'iscrizione di uno studente

Table 2: Dizionario delle Classi: end

Classe	Descrizione	Attributi
Sessione	Descrittore di ciascuna sessione presente in un corso	<b>oraInizio</b> (Data): orario di inizio della sessione. <b>numAderenti</b> (Int): numero di partecipanti alla sessione. <b>quantità</b> (Double): la quantità degli ingredienti per la ogni studente cosi da evitare sprechi. <b>teoria</b> (String): argomento teorico spiegato in sessione <b>tipoSessione</b> (TipoSessione): indica il tipo di sessione: (online) oppure (presenza).
Ricetta	Descrittore di ciascuna ricetta usata in una sessione.	<b>nome</b> (String): nome della ricetta. <b>descrizione</b> (String): la preaparazione della ricetta. <b>tempo</b> (Time): tempo per la preparazione della ricetta.
Ingrediente	Descrittore di ciascun ingrediente usato per una ricetta.	<b>nome</b> (String): nome dell'ingrediente. <b>unitàDiMisura</b> (String): specifica dell'unità di misura di un ingrediente ES: Kg, litri ecc.. .
Adesione	Classe associativa per gestire l'associazione tra Utente (studente/chefStudente) e Sessione (presenza).	<b>dataAdesione</b> (Data): data relativa all'adesione di uno studente ad una sessione.
Richiede	Classe associativa per gestire l'associazione tra Ingrediente e Ricetta.	<b>quantitàNecessaria</b> (Double): quantità utilizzata di ogni singolo ingrediente per una ricetta.
Categoria_Corso	Tabella per l'attributo multivalore categoria del corso.	<b>categoria</b> (String): una categoria associata a un corso.
Specializzazione_Chef	Tabella per l'attributo multivalore specializzazione dello chef.	<b>specializzazione</b> (String): una specializzazione associata a uno chef.
Gestisce	Classe associativa per gestire l'associazione tra Chef (Utente) e Corso.	Associazione tra <b>email_chef</b> e <b>id_corso</b> .
Prepara	Classe ponte per l'associazione tra Sessione e Ricetta.	Associazione tra <b>id_sessione</b> e <b>id_ricetta</b> .

### 3.2 Dizionario Associazioni

Table 3: Dizionario delle associazioni:

<b>Associazione</b>	<b>Descrizione</b>	<b>Molteplicità delle Classi coinvolte</b>
adesione	Richiesta di adesione da parte di uno studente	Utente [0..*] - Adesione [1]
iscrizione	Richiesta di iscrizione da parte di uno studente	Utente [0..*] - Iscrizione [1]
gestisce	Gestione di un corso da parte di uno chef	Utente [0..*] - Corso [1..*]
compone	Un corso è composto da più sessioni	Corso [1] - Sessione [1..*]
prepara	Ad ogni sessione in presenza si prepara una ricetta	Sessione [1..*] - Ricetta [1..*]
haBisogno	ogni ricetta ha bisogno di sapere la quantità degli ingredienti	Ricetta [1] - Richiede [1..*]
necessita	Un ingrediente necessita di sapere la quantità per la ricetta	Richiede [1..*] - Ingrediente [1]
per	Richiesta di adesione per una sessione in presenza	Adesione [1] - Sessione [0..*]
relativa	Iscrizione di uno studente per un corso	Iscrizione [1] - Corso [1..*]

### 3.3 Dizionario Vincoli

Table 4: Dizionario dei vincoli

Vincolo	Tipo	Descrizione
Email	Dominio	L'attributo "email" della classe Utente deve contenere una combinazione non nulla di lettere, numeri e simboli, seguiti da una chiocciola ("@"), altre lettere/simboli, un punto ((".")), e finire con almeno due lettere.
Password	Dominio	L'attributo "password" della classe Utente deve essere lungo almeno 8 caratteri, e avere almeno una lettera maiuscola, una minuscola, un numero e un carattere speciale.
Nome	Dominio	L'attributo "Nome" della classe Utente deve contenere solo stringhe di caratteri alfabetici, non vuote e con lunghezza $\leq 50$ caratteri.
Cognome	Dominio	L'attributo "Cognome" della classe Utente deve contenere solo stringhe di caratteri alfabetici, non vuote e con lunghezza $\leq 50$ caratteri.
TipoUtente	Dominio	L'attributo "tipoUtente" della classe Utente è un'enumerazione che può assumere solo i valori: chef, studente, chefStudente.
Specializzazione	N-tupla	Se l'attributo "TipoUtente" è chef o chefStudente, allora l'attributo "Specializzazione" assume un valore alfabetico con lunghezza $\leq 50$ .

Vincolo	Tipo	Descrizione
Matricola	N-tupla	Se l'attributo "TipoUtente" è studente o chefStudente, allora l'attributo "Matricola" assume un valore alfanumerico con lunghezza $\leq 50$ .
DataInizio	Dominio	L'attributo "DataInizio" della classe Corso deve essere una data valida nel formato previsto, non può essere nulla.
Nome	Dominio	L'attributo "Nome" della classe Corso deve contenere una stringa non vuota con solo caratteri alfabetici e lunghezza massima 50 caratteri.
Categoria	N-tupla	L'attributo "Categoria" della classe Corso deve contenere almeno una categoria. Ogni elemento deve essere una stringa non vuota.
Frequenza	Dominio	L'attributo "Frequenza" della classe Corso deve contenere una stringa che rappresenta un intervallo temporale coerente.
NumPartecipanti	Intrarelazionale	L'attributo "NumPartecipanti" della classe Corso è calcolato automaticamente in base al numero di iscrizioni associate.
NumSessioni	Dominio	L'attributo "NumSessioni" della classe Corso deve essere un intero positivo $>0$ .
OraInizio	Dominio	L'attributo "OraInizio" della classe Sessione deve essere una data valida nel formato previsto, non può essere nulla.

Vincolo	Tipo	Descrizione
TipoSessione	Dominio	L'attributo "TipoSessione" della classe Sessione è un'enumerazione che può assumere solo i valori: presenza, online.
Quantità	Intrarelazionale	Se l'attributo "TipoSessione" è "presenza", allora "Quantità" assume un valore numerico $>0$ .
NumAderenti	Intrarelazionale	Se l'attributo "TipoSessione" è "presenza", allora "NumAderenti" è calcolato automaticamente in base al numero di adesioni associate.
Teoria	Intrarelazionale	Se l'attributo "TipoSessione" è "online", allora "Teoria" deve essere una stringa alfanumerica con lunghezza $\leq 255$ .
Iscrizione	Ntupla	La relazione "Iscrizione" può coinvolgere solo utenti con tipoUtente = studente o chefStudente.
Adesione	Ntupla	La relazione "Adesione" può coinvolgere solo utenti con tipoUtente = studente o chefStudente e solo sessioni con tipoSessione = presenza.
Prepara	Ntupla	La relazione "Prepara" può coinvolgere solo sessioni con tipoSessione = presenza. Ogni sessione in presenza prepara almeno una ricetta e ogni ricetta è preparata in almeno una sessione.
Nome	Dominio	L'attributo "Nome" della classe Ricetta deve contenere una stringa alfabetica non vuota con lunghezza $\leq 50$ .



<b>Vincolo</b>	<b>Tipo</b>	<b>Descrizione</b>
Descrizione	Dominio	L'attributo "Descrizione" della classe Ricetta deve contenere una stringa alfabetica non vuota con lunghezza $\leq 250$ .
Tempo	Dominio	L'attributo "Tempo" della classe Ricetta deve contenere un orario (ore, minuti, secondi) valido e non nullo.
Nome	Dominio	L'attributo "Nome" della classe Ingrediente deve contenere una stringa alfabetica non vuota con lunghezza $\leq 50$ .
UnitàDiMisura	Dominio	L'attributo "UnitàDiMisura" della classe Ingrediente deve contenere una stringa alfabetica non vuota con lunghezza $\leq 50$ .

## 4 Schema Logico

### 4.1 Tabelle

Leggenda:

- Le parole con il **grassetto** sono indicate tutti i nomi delle tabelle
- Le parole sottolineate una volta indicano le chiavi primarie
- Le parole sottolineate due volte indicano le chiavi esterne
- Le parole in *italico* indicano che una chiave primaria è composta da più attributi

<b>Utente</b>	<u>email</u> , nome, cognome, password, matricola, tipoUtente
<b>Corso</b>	<u>id_corso</u> , dataInizio, nome, frequenza, numPartecipanti, numSessioni
<b>Sessione</b>	<u>id_sessione</u> , oraInizio, numAderenti, quantità, teoria, TipoSessione, <u>id_corso</u>
<b>Ricetta</b>	<u>id_ricetta</u> , nome, descrizione, tempo
<b>Ingrediente</b>	<u>nome</u> , unitàDiMisura
<b>Adesione</b>	dataAdesione, <i>email</i> , <i>id_sessione</i> , <u>email</u> , <u>id_sessione</u>
<b>Iscrizione</b>	dataIscrizione, stato, <i>email</i> , <i>id_corso</i> , <u>email</u> , <u>id_corso</u>
<b>Richiede</b>	quantitàNecessaria, <i>id_ricetta</i> , <i>nomeIngrediente</i> , <u>id_ricetta</u> , <u>nomeIngrediente</u>
<b>Specializzazione_Chef</b>	<i>email_chef</i> , <i>specializzazione</i> , <u>email_chef</u>
<b>Categoria_Corso</b>	<i>id_corso</i> , <i>categoria</i> , <u>id_corso</u>
<b>Gestisce</b>	<i>email_chef</i> , <i>id_corso</i> , <u>email_chef</u> , <u>id_corso</u>
<b>Prepara</b>	<i>id_sessione</i> , <i>id_ricetta</i> , <u>id_sessione</u> , <u>id_ricetta</u>

## Elenco delle tabelle

1	Dizionario delle Classi:1 . . . . .	11
2	Dizionario delle Classi: end . . . . .	12
3	Dizionario delle associazioni: . . . . .	13
4	Dizionario dei vincoli . . . . .	14