



# UninaFoodLab

Antonino De Martino  
N86005103

Simone Barbella  
N86004906

Gruppo  
OOBD57

Anno Accademico 2024/2025

# Indice

<b>1</b>	<b>Descrizione del progetto</b>	<b>3</b>
<b>2</b>	<b>Funzionalità</b>	<b>4</b>
2.1	Funzionalità richieste . . . . .	4
2.2	Funzionalità aggiuntive . . . . .	4
<b>3</b>	<b>Progettazione concettuale</b>	<b>6</b>
3.1	Class Diagram . . . . .	6
<b>4</b>	<b>Legenda</b>	<b>7</b>
4.1	Guida alla Lettura dei File Java . . . . .	7
<b>5</b>	<b>Repository</b>	<b>8</b>

# 1 Descrizione del progetto

Il sistema di riferimento, denominato **UninaFoodLab**, richiede la creazione di una piattaforma gestionale integrata dedicata **all'organizzazione e al coordinamento di attività laboratoriali in ambito culinario**. Il sistema è progettato per centralizzare la gestione delle risorse didattiche, permettendo una sinergia tra la pianificazione dei corsi e l'esecuzione pratica delle sessioni di cucina.

Le interazioni all'interno della piattaforma sono regolate in modo che gli utenti, identificati come **Chef**, possano ricoprire ruoli gestionali dinamici: dalla creazione di nuovi percorsi formativi (**Corsi**) alla supervisione delle singole attività in cucina. Gli **Chef** hanno la possibilità di popolare il sistema con contenuti tecnici, quali **ricette dettagliate** e **liste di ingredienti**, garantendo la coerenza dei dati necessari per ogni sessione.

Le componenti didattiche sono suddivise in base alla tipologia di attività prevista:

- **Corsi:** rappresentano l'unità formativa principale, caratterizzata da una descrizione e una programmazione temporale;
- **Sessioni Pratiche:** costituiscono l'attuazione concreta del corso, all'interno delle quali vengono coordinate le ricette da eseguire e i relativi ingredienti necessari.
- **Notifiche:** fungono da strumento di comunicazione interna per informare gli utenti su aggiornamenti relativi a corsi o avvisi di sistema.

Il sistema gestisce quindi l'intero ciclo di vita di una sessione didattica: dalla sua pianificazione iniziale e configurazione dei materiali (**ingredienti** e **ricette**), passando per l'autenticazione sicura del personale addetto, fino alla conclusione della **sessione** e alla **notifica** agli utenti interessati.

## 2 Funzionalità

### 2.1 Funzionalità richieste

Le funzionalità richieste dal dominio per la gestione del laboratorio culinario e successivamente implementate sono di seguito definite:

- **Autenticazione** utente tramite credenziali fornite in fase di login;
- **Gestione** delle sessioni pratiche, con possibilità di inserimento e coordinamento delle attività didattiche;
- **Pianificazione** dei corsi di cucina, comprensiva di descrizione e dettagli organizzativi;
- **Creazione e gestione** di ricette dettagliate da associare alle sessioni di lavoro;
- **Inserimento e gestione** degli ingredienti necessari per ogni singola ricetta o sessione;
- **Sistema di messaggistica interna** per l'invio e la ricezione di notifiche tra gli Chef e lo staff;
- **Visualizzazione** dei dettagli delle sessioni e delle relative risorse assegnate;
- **Dashboard riepilogativa** per il monitoraggio dello stato delle attività e delle comunicazioni ricevute;
- **Generazione** di report sintetici che mostrino le statistiche sull'andamento dei corsi e rappresentazioni grafiche dei dati.

### 2.2 Funzionalità aggiuntive

Le funzionalità aggiuntive, non espressamente richieste ma implementate per garantire un'esperienza d'uso più completa e fluida, sono definite di seguito:

- **Navigazione dinamica** tra le scene mediante utility centralizzata per un passaggio fluido tra i vari moduli del sistema;
- **Sistema di alert e feedback visivi** per confermare il successo delle operazioni di inserimento o segnalare errori di compilazione;
- Filtro e ricerca delle sessioni pratiche per facilitare il reperimento di specifiche attività nel tempo;

- **Visualizzazione** del profilo utente (Chef) con le rispettive informazioni anagrafiche e professionali;
- **Logout** dalla sessione corrente per garantire la sicurezza dell'accesso post-utilizzo.;
- **Gestione automatizzata** dei Coordinator per la separazione logica delle sezioni all'interno di interfacce complesse;
- **Accesso rapido** alle informazioni aggiuntive di ricette e corsi per una consultazione immediata dei dettagli tecnici.

## 3 Progettazione concettuale

### 3.1 Class Diagram

La progettazione del sistema FoodLab è stata trasposta in un Class Diagram che ne descrive la struttura statica, le gerarchie di ereditarietà e le relazioni di dipendenza tra i moduli. La modellazione segue rigorosamente il **pattern BCE**, distinguendo nettamente le interfacce grafiche dalla logica di business e dalla persistenza dei dati.

**Nota sulla visualizzazione:** Data l'elevata densità di informazioni e la complessità del Class Diagram, si è scelto di non includerlo direttamente nel presente documento per preservarne la leggibilità. La versione integrale è consultabile nel link sottostante.

<https://ibb.co/6Jc0Wsks> Class Diagram

Descrizione sintetica del diagramma:

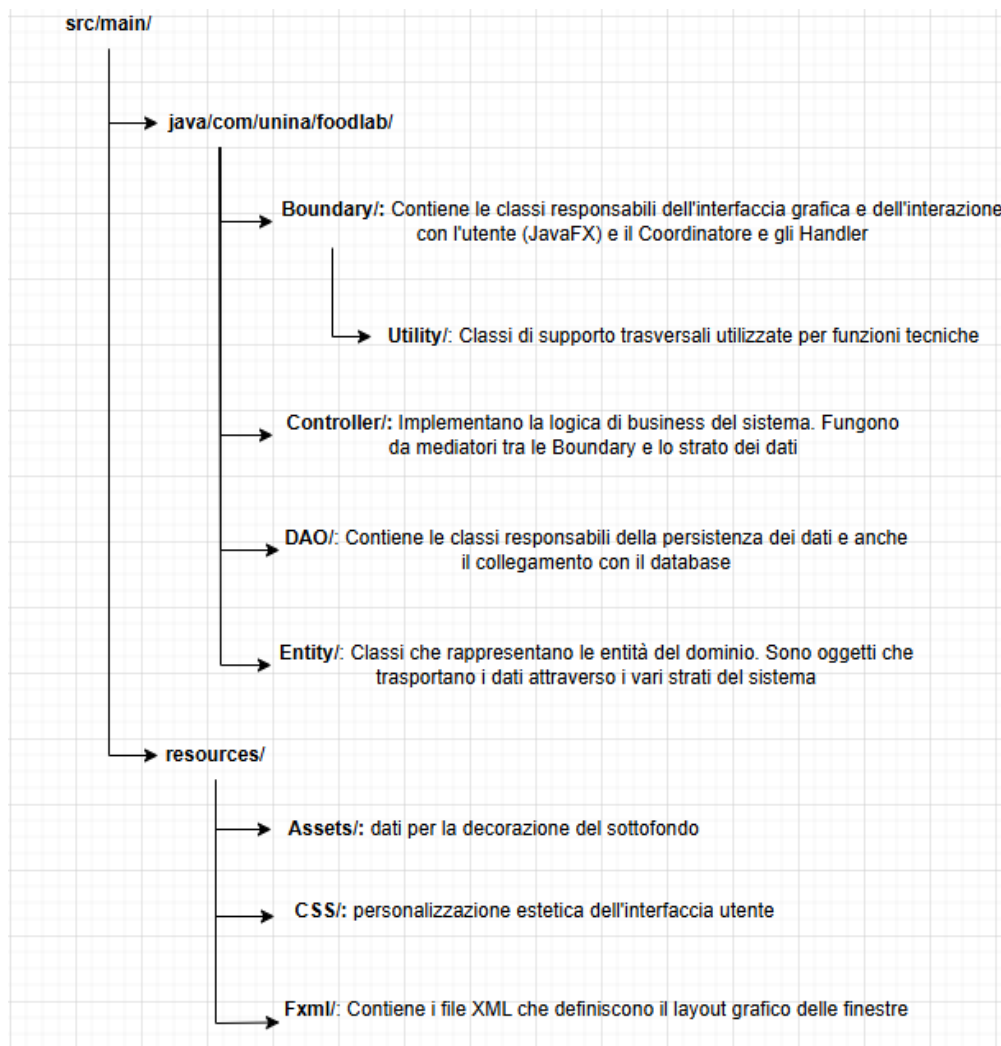
- **Livello Presentation (Boundary):** Comprende le classi che gestiscono l'interazione con l'utente e il caricamento dei file FXML, supportate da classi Coordinator e Handler per la gestione di sezioni complesse;
- **Livello Application (Control):** Centralizzato nei "Gestori", i quali coordinano i flussi di dati tra le interfacce e il database, garantendo l'integrità delle operazioni;
- **Livello Data (Entity e DAO):** Include la definizione degli oggetti del dominio e i relativi Data Access Object per la comunicazione con il database;
- **Supporto Tecnico (Utility):** Classi trasversali dedicate alla navigazione, alla formattazione e alla gestione degli alert grafici.

## 4 Legenda

### 4.1 Guida alla Lettura dei File Java

Per facilitare la comprensione dell'implementazione e della struttura del codice sorgente, viene di seguito riportata una legenda descrittiva delle directory e del loro contenuto, seguendo l'organizzazione del progetto basata sul pattern BCE:

Figure 1: Struttura delle directory del progetto FoodLab



## 5 Repository

Per la gestione del codice sorgente e la tracciabilità dello sviluppo è stato utilizzato il sistema di controllo di versione Git. La repository completa del progetto è consultabile al seguente indirizzo:

<https://github.com/SimoneBarbella/Progetto-Basi-Object>