

UNIVERSITÀ DEGLI STUDI DI NAPOLI
PARTHENOPE
DIPARTIMENTO DI INGEGNERIA



**CORSO DI LAUREA IN INGEGNERIA E SCIENZE
INFORMATICHE PER LA CYBERSECURITY**

TITOLO PROGETTO

UniLabManager

DOCENTE

PROFESSORE DANIELE GRANATA

STUDENTE

Andrea Di Palo

ANNO ACCADEMICO 2024/2025

Descrizione del progetto:

Il progetto *UniLabManager* è un sistema web progettato per gestire in maniera centralizzata le attività didattiche e sperimentali svolte all'interno dei laboratori universitari.

L'obiettivo principale è offrire un'interfaccia moderna ed efficiente per la **creazione e supervisione di progetti scientifici**, la **gestione di esperimenti**, e la **prenotazione di attrezzature e laboratori**, con ruoli differenziati per **professori, studenti e tecnici**.

Il sistema è stato realizzato per simulare un contesto reale accademico, in cui diversi attori interagiscono in un ambiente strutturato. La progettazione tiene conto sia della flessibilità di utilizzo sia delle regole di integrità dei dati, con l'ausilio del framework Django.

Obiettivi:

- Permettere la **registrazione differenziata** degli utenti secondo il proprio ruolo (Professore, Studente, Tecnico)
- Consentire al Professore di creare progetti sperimentali e associarvi esperimenti
- Fornire la possibilità di **prenotare attrezzature e laboratori** per ogni esperimento
- Offrire una **dashboard personalizzata** per ciascun utente
- Garantire **sicurezza dei dati**, autenticazione tramite matricola e accessi controllati

Requisiti Funzionali

RF1 – Registrazione dell'utente

Il sistema deve permettere la registrazione di nuovi utenti specificando il ruolo (Professore, Studente, Tecnico), tramite un form personalizzato per ciascun tipo. Durante la registrazione, viene verificata e salvata una matricola unica con formato specifico (P12345 per professori, S12345 per studenti, T12345 per tecnici).

RF2 – Autenticazione

Il sistema deve consentire l'autenticazione degli utenti registrati tramite matricola e password.

RF3 – Accesso a dashboard personalizzata

Dopo l'autenticazione, ogni utente accede ad una dashboard diversa a seconda del ruolo:

- I professori visualizzano i propri progetti sperimentali e possono crearne di nuovi.
- Gli studenti accedono a contenuti didattici o prenotazioni (in futuro).
- I tecnici visualizzano le prenotazioni da gestire.

RF4 – Creazione progetto sperimentale (Professore)

Un professore autenticato può creare nuovi progetti sperimentali inserendo titolo, descrizione, obiettivi, date, ecc.

RF5 – Visualizzazione ed eliminazione dei progetti

I professori possono visualizzare l'elenco dei propri progetti e scegliere di eliminarne uno se necessario.

RF6 – Creazione esperimenti associati a un progetto

Per ogni progetto, il professore può creare uno o più esperimenti, specificando descrizione, obiettivi, materiali e date.

RF7 – Prenotazione laboratorio e attrezzature

Durante la creazione di un esperimento, il sistema consente al professore di prenotare un laboratorio e selezionare una o più attrezzature, evitando conflitti di orario o doppie prenotazioni.

RF8 – Verifica disponibilità risorse

Il sistema deve impedire la prenotazione di laboratori o attrezzature già occupati nella stessa fascia oraria.

RF9 – Partecipazione agli esperimenti (Studente)

Il sistema deve consentire agli **studenti autenticati** di visualizzare l'elenco dei **progetti disponibili** e di **richiedere la partecipazione** agli esperimenti collegati. L'accesso a queste funzionalità è riservato solo agli utenti con ruolo "studente" che hanno effettuato il login.

RF10 – Gestione limite partecipanti per esperimento

Ogni esperimento deve avere un campo che indica il **numero massimo di studenti ammessi**. Il sistema deve impedire ulteriori iscrizioni quando il limite è stato raggiunto

RF11 – Logout

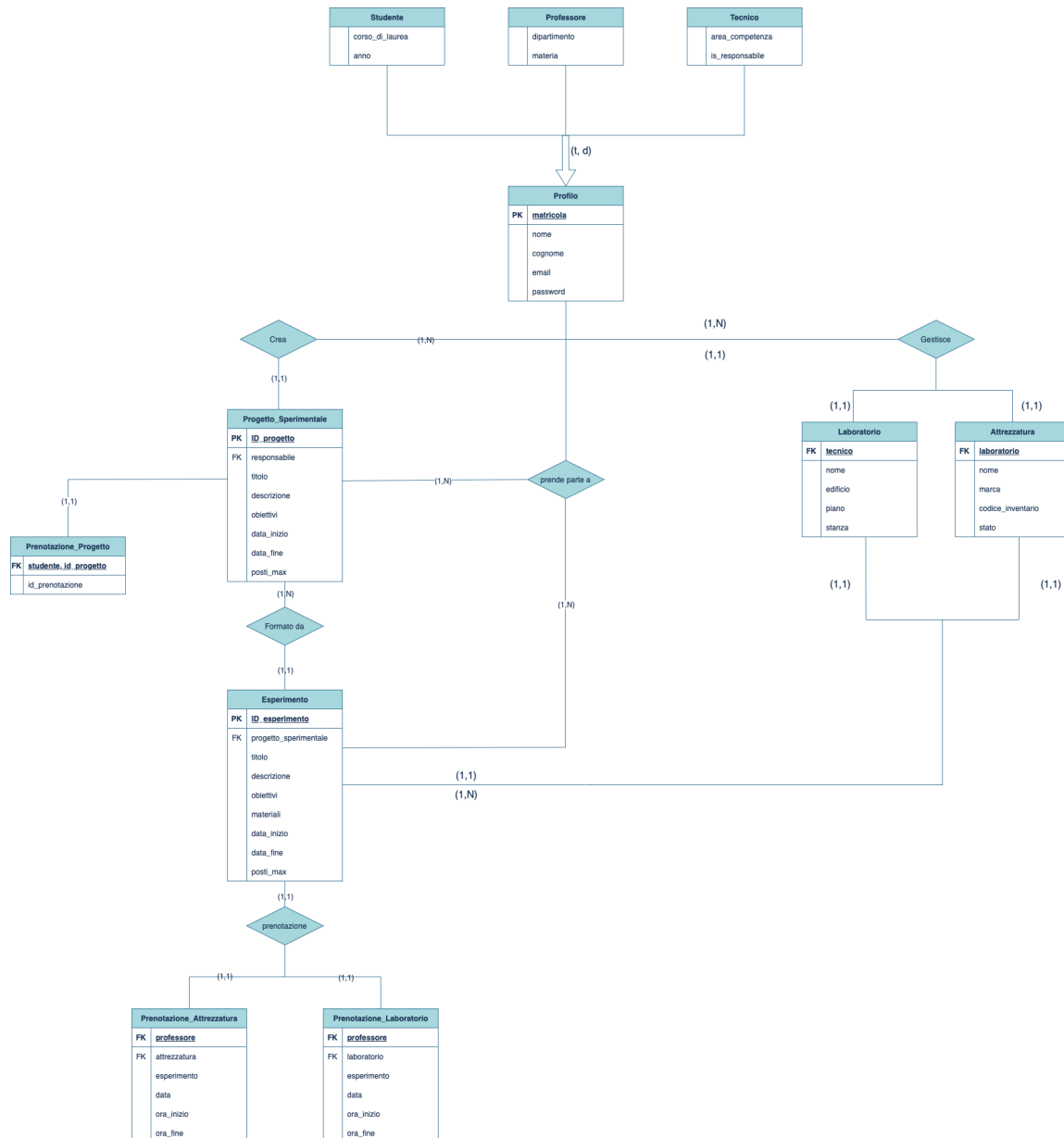
Il sistema deve permettere in ogni momento il logout dell'utente.

Modello Concettuale (E-R)

Il servizio web è pensato per la gestione dei progetti e delle attività sperimentali nei laboratori universitari.

1. I **professori** possono creare progetti, definire esperimenti e prenotare laboratori e attrezzature.
2. Gli **studenti**, una volta autenticati, possono visualizzare i progetti disponibili, prenotarsi e monitorare le attività a cui partecipano.
3. I **tecnici** hanno accesso a un pannello dedicato per gestire la disponibilità delle attrezzature e supervisionare le prenotazioni.

Ogni utente accede con credenziali univoche e visualizza solo le funzionalità legate al proprio ruolo. Il sistema garantisce un'organizzazione efficiente, evitando sovrapposizioni e semplificando la comunicazione tra studenti, docenti e tecnici.



Inizialmente è stato realizzato un modello E/R con un'entità padre e tre entità figlie, collegate da una relazione di specializzazione totale e disgiunta.

1. Relazione Profilo → Professore & Studente & Tecnico
2. Vincolo di copertura: Totale → Ogni Profilo è sempre un Professore, uno Studente o un Tecnico.
3. Vincolo di disgiunzione: Disgiunta → Un Profilo è o Professore o Studente o Tecnico, mai una combinazione delle tre.

La specializzazione dell' entità Profilo è stata definita in questo modo:

- Attributi comuni: matricola, nome, cognome, email
- Attributi privati: password

Soluzione alla generalizzazione

Accorpamento in padre + estensione specializzata

Nel progetto, si è scelto di adottare un'architettura che combina una **generalizzazione in una classe padre** (Profilo) con **specializzazioni distinte** (Studente, Professore, Tecnico) per modellare gli utenti del sistema. Questa scelta è stata presa in modo da:

1. **Evitare ridondanza:**

La classe **Profilo** centralizza i dati comuni a tutte le tipologie di utente (es. matricola, ruolo, utente Django), evitando duplicazioni e semplificando le operazioni di login, autenticazione, controllo ruolo.

2. **Consentire estensioni specializzate:**

Le classi **Studente**, **Professore** e **Tecnico** ereditano logicamente da **Profilo** e permettono di gestire informazioni specifiche (es. corso di laurea, dipartimento, area di competenza) senza appesantire la struttura del modello base. Questo consente di separare in modo chiaro la logica e le funzionalità legate a ciascun ruolo.