A.s. 2024/25 Verifica del 10/02/2025

Classe 5IC Docenti: Infante G. , Torsello M.

Informatica Studente: David Anvar

Legenda:  
Entità

Relazioni

Attributi

Cardinalità

**Progettazione di una applicazione WEB**

**TRACCIA:**

Una biblioteca comunale necessita di un sistema informativo web, tramite cui gli utenti possono visualizzare e prenotare libri, mentre solo gli amministratori possono aggiungere, modificare o eliminare libri. Il sistema, inoltre, deve essere in grado di distinguere tra utenti già conosciuti e utenti che visitano il sito per la prima volta.

**IPOTESI:**

Visto che solo l’amministratore ha la possibilità di modificare i libri la pagina di accesso per la modifica dei libri l’ha applico solo se inserisce le credenziali di login. Suppongo che gli amministratori sono già registrati perciò è necessario un solo login. Se gli amministratori devono accedere alla tabella, inserisce un codice che che solo gli amministratori conoscono. Visto che la quantità non è specificata del numero di amministratori applico un attributo in più che è un codice di accesso per la tabella

Sappiamo che se un utente è già conosciuto all’interno della biblioteca faccio il login mentre se non è registrato e fa cmq l’accesso mi riporta sulla pagina di registrazione.

La relazione prenotare può essere considerata non solo come relazione ma come entità: conoscendo quindi: la data e ora di pubblicazione e il luogo.

Il Tipo di cliente indica se è un utente o un amministratore e le creo tramite menù a tendina

CARDINALITA’:

Se un libro viene prenotato da una sola persona, il libro non può essere prenotato da più persone a meno che il libro non sia uguale, ma se consideriamo lo stesso libro e che ci sia un solo libro in tutta la biblioteca allora la prenotazione non può essere effettuata.

**Schema E-R(Entità-Relazioni):**

**Entità e Attributi:**

**Cliente:**

-id\_cliente (PK)

-nome

-cognome

-data\_di\_nascita

-email

-CF

-tipo\_di\_cliente

**Libro:**  
-id\_libro (PK)

-titolo

-data\_di\_pubblicazione

-prezzo

-descrizione

-volume

-luogo\_di\_pubblicazione

**Prenotazione:**

id\_prenotazione (PK)

Data\_di\_prenotazione

Ora\_di\_prenotazione

luogo\_di\_prenotazione

**Relazioni:**

Cliente-Libro:

La relazione tra Cliente e Libro è prenotare:

Il Cliente può prenotare almeno un libro, ma può prenotare più libri (1:N)

Un Libro può essere prenotato da un solo cliente (1:1)

Cliente-Prenotazione:

La relazione tra Cliente e Prenotazione è effettuare:

Il Cliente può effettuare più prenotazioni (N:M)

La Prenotazione può essere effettuata da un solo cliente (1:1)

**Diagramma:**

**Schema logico:**  
Tabella Cliente (id\_cliente (PK), nome, cognome, data\_di\_nascita, email, CF, tipo\_di\_cliente);

Tabella libro (id\_libro (PK), titolo, data\_di\_pubblicazione, prezzo, descrizione, volume, luogo\_di\_pubblicazione);

Tabella Prenotazione (id\_prenotazione (PK), data\_di\_prenotazione, ora\_di\_prenotazione, luogo\_di\_prenotazione);

Tabelle relazionali:  
Tabella Cliente-Libro (id\_cliente, id\_libro, )

Tabella Cliente-Prenotazione (id\_cliente, id\_ libro, )

**QUERY:**CREATE TABLE Cliente (

ID\_cliente INT(10) PRIMARY KEY ,

nome VARCHAR(80) NOT NULL ,

cognome VARCHAR(80) NOT NULL ,

data\_di\_nascita INT(20) NOT NULL ,

CF VARCHAR(16) NOT NULL ,

email VARCHAR(100) NOT NULL ,

tipo\_di\_cliente INT(15) NOT NULL

);

CREATE TABLE Libro (

ID\_libro INT(10) PRIMARY KEY ,

titolo VARCHAR(80) NOT NULL ,

data\_di\_pubblicazione INT(20) NOT NULL ,

prezzo INT(20) NOT NULL ,

descrizione VARCHAR(16) NOT NULL ,

volume VARCHAR(100) NOT NULL ,

luogo\_di\_pubblicazione INT(15) NOT NULL

);