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

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

Informatica Studente: David Anvar

**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.

Dall’entità Libro nasce un’altra entità: Autore, in cui indica che l’autore pubblica libri e quindi indico la sua cardinalità. Chiaramente per maggiore chiarezza e organizzazione degli attributi, l’entità Autore non esiste nel testo, ma visto che fa parte dell’entità Libro ho creato una entità apposita per non avere troppi attributi nell’entità libro.

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

-metodo\_di\_pagamento

**Autore:**

-id\_autore (PK)

-nome

-cognome

-data\_di\_nascita

-nazionalità

**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 più clienti, ma un cliente può prenotare più libri (N:M)

**Cliente-Prenotazione:**

La relazione tra Cliente e Prenotazione è **effettuare**:

Un Cliente può effettuare più prenotazioni (1:N)

Una Prenotazione può essere effettuata da più clienti (1:N)

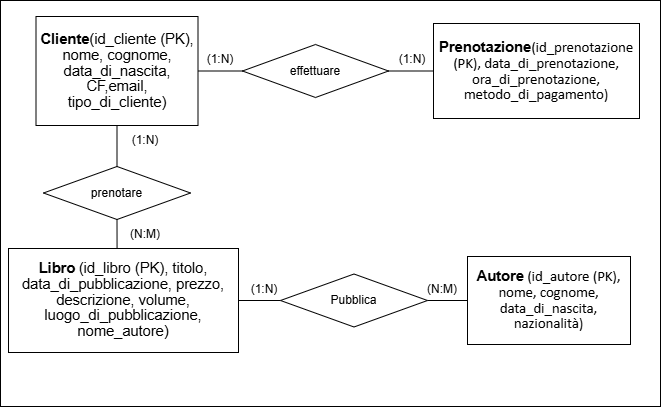
**Libro-Autore:**

La relazione tra Libro e Autore è **pubblicare**:

Un Libro può essere pubblicato da un solo autore(1:N)

Un Autore può pubblicare almeno un libro, ma può pubblicare più libri (N:M)

**Diagramma E/R:**

****

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

**Tabella Libro** (id\_libro (PK), titolo, data\_di\_pubblicazione, prezzo, descrizione, volume, luogo\_di\_pubblicazione, nome\_autore);

**Tabella Prenotazione** (id\_prenotazione (PK), data\_di\_prenotazione, ora\_di\_prenotazione, metodo\_di\_pagamento);

**Tabella Autore** (id\_autore (PK), nome, cognome, data\_di\_nascita, nazionalità);

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

Tabella Cliente-Prenotazione (id\_cliente FK, id\_prenotazione FK);

Tabella Libro-Autore (id\_libro FK, id\_autore FK);

**Schema Relazionale:**

1) **Cliente:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ID\_cliente** | **nome** | **cognome** | **data\_di\_nascita** | **CF** | **email** | **tipo\_di\_cliente** |
| 01 | Mario | Rossi | 13/11/2001 | SLKDDC32CXVDFG | Mario.R@gmail.com | Amministratore |

2) **Libro:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID\_libro** | **titolo** | **Data\_di\_pubblicazione** | **prezzo** | **descrizione** | **volume** | **nome\_autore** | **Luogo\_di\_pubblicazione** |
| 01 | Piccolo Principe | 10/12/1990 | 10£ | È un libro di fantasia | A | Laura Bianchi | Francia |

3) **Prenotazione:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID\_prenotazione** | **Data\_di\_prenotazione** | **ora\_di\_prenotazione** | **metodo\_di\_pagamento** |
| 01 | 10/12/2024 | 12:36 | PayPal |

4) **Autore:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID\_autore** | **nome** | **cognome** | **data\_di\_nascita** | **nazionalità** |
| 01 | Alessandro | Manzoni | 07/03/1785 | Italiano |

**QUERY:**

**Tabella Cliente:**CREATE TABLE cliente (

ID\_cliente INT AUTO\_INCREMENT 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 UNIQUE,

password VARCHAR(100) NOT NULL,

tipo\_di\_cliente TINYINT(1) NOT NULL

);

**Tabella Libro:**

CREATE TABLE libro(

ID\_libro INT AUTO\_INCREMENT PRIMARY KEY ,

titolo VARCHAR(80) NOT NULL ,

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

prezzo INT(20) NOT NULL ,

descrizione VARCHAR(100) NOT NULL ,

volume VARCHAR(16) NOT NULL ,

luogo\_di\_pubblicazione VARCHAR(15) NOT NULL ,

nome\_autore VARCHAR(100) NOT NULL

);

**Tabella Autore:**

CREATE TABLE autore(

ID\_autore INT AUTO\_INCREMENT PRIMARY KEY ,

nome VARCHAR(80) NOT NULL ,

cognome VARCHAR(80) NOT NULL ,

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

nazionalita VARCHAR(50) NOT NULL

);

**Tabella relazione Cliente-Libro:**

CREATE TABLE cliente\_libro (

ID\_cliente INT,

ID\_libro INT,

PRIMARY KEY (ID\_cliente, ID\_libro),

FOREIGN KEY (ID\_cliente) REFERENCES cliente(ID\_cliente),

FOREIGN KEY (ID\_libro) REFERENCES libro(ID\_libro)

);

**Tabella relazione Cliente-Prenotazione:**

CREATE TABLE cliente\_prenotazione (

ID\_prenotazione INT AUTO\_INCREMENT PRIMARY KEY,

email\_cliente VARCHAR(255),

ID\_libro INT, -- Usa l'ID del libro come chiave esterna

data\_di\_prenotazione DATE NOT NULL,

ora\_di\_prenotazione TIME NOT NULL,

data\_di\_scadenza DATE NOT NULL,

metodo\_di\_pagamento VARCHAR(50) NOT NULL,

FOREIGN KEY (email\_cliente) REFERENCES cliente(email),

FOREIGN KEY (ID\_libro) REFERENCES libro(ID\_libro) -- Relazione con ID\_libro

);

**Tabella relazione Libro-Autore:**

CREATE TABLE libro\_autore (

ID\_libro INT,

ID\_autore INT,

PRIMARY KEY (ID\_autore, ID\_libro),

FOREIGN KEY (ID\_libro) REFERENCES autore(ID\_autore) ,

FOREIGN KEY (ID\_autore) REFERENCES libro(ID\_libro)

);

**Query dati (Inserimento dati delle tabelle-relazionali):**

INSERT INTO Cliente (ID\_cliente, nome, cognome, data\_di\_nascita, CF, email, tipo\_di\_cliente)

VALUES (01, 'Mario', 'Rossi', 13/11/2001, 'SLKDDC32CXVDFG', 'Mario.R@gmail.com', 'Amministratore');

INSERT INTO Libro ('ID\_libro', 'titolo', 'Data\_di\_pubblicazione', 'prezzo', 'descrizione', 'nome\_autore', 'luogo\_di\_pubblicazione')

VALUES (01, 'Piccolo Principe', 10/12/1990, 10, 'E’ un libro di fantasia', 'A', 'Laura Bianchi', 'Francia');

INSERT INTO Prenotazione ('ID\_prenotazione', 'Data\_di\_prenotazione', 'ora\_di\_prenotazione', 'metodo\_di\_pagamento')

VALUES (01, 'Rossi', 10/12/2024, 12:36, 'PayPal');

INSERT INTO Autore (01 ,'nome', 'cognome', 'data\_di\_nascita', 'nazionalità')

VALUES (01, 'Alessandro', 'Manzoni', 07/03/1785, 'italiano');

INSERT INTO Cliente-Libro (id\_cliente, id\_libro)

VALUES (1, 1);

INSERT INTO Cliente-Prenotazione (id\_cliente, id\_prenotazione)

VALUES (1, 1);

INSERT INTO Libro-Autore (id\_libro, id\_autore)

VALUES (1, 1);

**Progettazione interfaccia utente tramite pagine web**