2024

Progetto Basi di Dati

Base di dati per la gestione di un negozio di libri

DAVIDE NICOLÒ [IN0501112]

1. Introduzione

Il progetto ha come obbiettivo quello di sviluppare un database relazionale per la gestione di un negozio di libri.

Questo database vorra’ essere un sistema di gestione in grado di aiutare un negozio nelle attivita quotidiane quali: tenere traccia dell’inventario, le informazioni sui clienti e degli ordini e dei dettagli di questi garantendo che tutte le operazioni siano eseguite in modo efficiente ed organizzato.

Il sistema di gestione del negozio consentira’ di gestire l’inventario dei libri in modo efficiente, permettendo l’inserimento di nuovi libri con dettagli come titolo, autore, prezzo, genere, anno di pubblicazione e quantita’ disponibile. Sara’ progettato in modo da facilitare gli aggiornamenti delle informazioni dei libri esistenti, cambiamenti di prezzo o nelle quantita presenti nell’inventario e operazioni come rimuovere i libri non piu’ disponibili dopo la vendita.

Per quanto riguarda la gestione dei clienti, verra’ fatta in modo semplice con la possibilita’ di registrare nuovi acquirenti con le loro informazioni persoanli come nome, cognome, indirizzo, numero di telefono ed email. Sara’ anche possibile aggiornare le informazioni facendo si che queste siano sempre accurate ed aggiornate rimuovendo i clienti non piu attivi.

Nella gestione degli ordini, il sistema permettera’ di registrare nuovi ordini effettuati dai clienti rendendo disponibili dettagli come la data in cui si e’ effettuato l’ordine ed il totale dell’ordine. Sara’ poi possibile gestire i dettagli di questi, ovvero i libri specifici inclusi in ogni ordine, la quantita’ ordinata e il prezzo per unita’. Inoltre si vuole che il sistema consenta di verificare lo stato degli ordini e tracciare la loro progressione dal momento in cui viene effettuato fino alla consegna.

Infine, il sistema offrira’ funzionalita’ di ricerca, permettendo di cercare libri nel database per titolo, autore, genere, o altri criteri per facilitare l’assistenza ai clienti e la gestione dell’inventario. Sara’ anche possibile verificare la disponibilita’ dei libri per assicurare che i clienti possano ordinare solo libri effettivamente disponibili.

* 1. Operazioni da effettuare sul database
* Inserimento di nuovi libri: Aggiungere nuovi libri all’inventario del negozio.
* Aggiornamento dei dettagli dei libri: Modificare le informazioni esistenti sui libri all’interno del database, come titolo, autore e prezzo.
* Eliminazione dei libri: Rimuovere libri dall’inventario del negozio.
* Registrazione dei nuovi clienti : Aggiungere nuovi clienti al sistema.
* Aggiornamento dei dettagli dei clienti: Modificare le informazioni esistenti sui clienti, come nome, indirizzo e numero di telefono.
* Eliminazione dei clienti: Rimuovere i clienti che non sono piu attivi
* Gestione degli ordini: Registrare gli ordini effettuati dai clienti.
* Gestione dei dettagli degli ordini: Registrare i dettagli dei libri inclusi all’interno degli ordini.
* Ricerca libri: Avere la possibilita di ricercare i libri per titolo, genere o autore.
* Verificare la disponibilita’ di un libro: Controllare se un libro e’ disponibile in magazzino.

1. Schema concettuale

Si e’ progettato dunque il seguente schema concettuale:

A diagram of a data flow

Description automatically generated

E’ importante specificare che essendo le chiavi primarie (PK) attributi che identificano in modo univoco ogni istanza di entita’, nello schema ad ogni entita’ e’ associato un attributo ID che funge da chiave primaria. Se appare l’attributo “ID” ma questo non coincide con il nome dell’entita’ allora sara’ una chiave esterna (FK).

* 1. Dizionario dei Dati

**Entita’:**

|  |  |  |  |
| --- | --- | --- | --- |
| Entita’ | Descrizione | Attributi | Identificatore Primario |
| Cliente | Cliente che effettua acquisti all’interno del negozio. | ID Cliente, Nome, Cognome, Indirizzo, Telefono, Email | ID Cliente |
| Ordine | Transazione di acquisto effettuata da un cliente. | ID Ordine, Data Ordine, Totale Ordine, ID Cliente | ID Ordine |
| Dettaglio Ordine | Dettaglio degli articoli inclusi in un ordine. | ID Dettaglio, Quantita’, Prezzo, ID Ordine, ID Libro | ID Dettaglio |
| Libro | Libro disponibile per la vendita nel negozio. | ID Libro, Titolo, Autore, Prezzo, Genere, Anno Pubblicazione, Quantita Disponibile | ID Librio |

**Relazioni:**

|  |  |  |  |
| --- | --- | --- | --- |
| Relazione | Descrizione | Entita’ Coinvolte | Attributi |
| Effettuata | Un cliente effettua uno o piu’ ordini | Cliente, Ordine | ID Cliente, ID Ordine |
| Contiene | Un ordine contiene uno o piu’ dettagli | Ordine, Dettaglio Ordine | ID Ordine, ID Dettaglio |
| Include | Un dettaglio di un ordine include un libro | Dettaglio ordine, Libro | ID Dettaglio, ID Libro |

1. Ristrutturazione dello schema concettuale

L’obbiettivo e’ quello di garantire una strutturea efficace e priva di ridondanze, con la possbilita’ di aggiungere ulteriori entita’ e relazioni per migliorare la gestione del sistema.

**Analisi delle ridondanze entita’ e relazioni**:

Non sono state rilevate ridondanze.

**Aggiunte/eliminazioni di entita’ e relazioni**:

Dopo aver effettuato l’analisi, non ci sono entita’ o relazioni che necessitano di eliminazione. Tutte le entita’ e relazioni definite sono necessarie per la gestione completa del sistema di negozio di libri.

Tuttavia per migliorare la gestione del negozio di libri e fornire ulteriori funzionalita’ potremmo considerare le seguenti aggiunte:

Entita’:

* Fornitore: per tracciare il fornitore dei libri

Attributi: ID Fornitore, Nome, Indirizzo e telefono

* Categoria: per classificare i libri

Attributi: Id Categoria e Nome

* Recensione: per permettere ai clienti di lasciare recensioni sui libri

Id Recensione, Id Cliente, Id Libro, Testo e Valutazione

Relazioni:

* Fornisce:Un fornitore puo’ fornire uno o piu libri
* Classifica: Una categoria puo’ classificare uno o piu’ libri
* Scrive: Un cliente puo scrivere una o piu’ recensioni
  1. Schema concettuale ristrutturato

Lo schema concettuale viene dunque ristrutturato come segue:

A diagram of a company

Description automatically generated

1. Schema Logico

Lo schema logico derivato e’ il seguente:

A computer screen shot of a computer

Description automatically generated

1. Analisi delle forme normali

* **Prima forma normale** (1NF) : Viene richiesto che i dati siano organizzate affinche ogni colonna contenga valori unici ed atomici.

Tutte le tabelle del database soddisfano i requisiti della *prima forma normale.*

* **Seconda forma normale** (2NF): Viene soddisfatta la 1NF e tutti gli attributi non chiave sono completamente dipendenti dalla chiave primaria.

Tutte le tabelle del database soddisfano i requisiti della *seconda forma normale*.

* **Terza forma normale** (3NF): Viene soddisfatta la 2NF e ogni attributo non chiave dipende solo dalla chiave primaria.

Tutte le tabelle del database soddisfano i requisiti della *terza forma normale.*

1. Sql

CREATE DATABASE NegozioLibri;

CREATE TABLE Cliente (

ID\_Cliente INT PRIMARY KEY,

Nome VARCHAR(100),

Cognome VARCHAR(100),

Indirizzo VARCHAR(255),

Telefono VARCHAR(20)

Email VARCHAR(100)

);

CREATE TABLE Ordine (

ID\_Ordine INT PRIMARY KEY,

Data\_Ordine DATE,

Totale\_Ordine DECIMAL(10, 2),

ID\_Cliente INT,

FOREIGN KEY (ID\_Cliente) REFERENCES Cliente(ID\_Cliente)

);

CREATE TABLE Dettaglio\_Ordine (

ID\_Dettaglio INT PRIMARY KEY,

Quantia’ INT,

Prezzo DECIMAL(10, 2),

ID\_Ordine INT,

ID\_Libro INT,

FOREIGN KEY (ID\_Ordine) REFERENCES Ordine(ID\_Ordine),

FOREIGN KEY (ID\_Libro) REFERENCES Libro(ID\_Libro);

);

CREATE TABLE Libro (

ID\_Libro INT PRIMARY KEY,

Titolo VARCHAR(255),

Autore VARCHAR(255),

Prezzo DECIMAL(10, 2),

Genere VARCHAR(100),

Anno\_Pubblicazione INT,

Quantita’\_Disponibile INT,

ID\_Fornitore INT,

FOREIGN KEY (ID\_Fornitore) REFERENCES Fornitore(ID\_Fornitore)

);

CREATE TABLE Fornitore (

ID\_Fornitore INT PRIMARY KEY,

Nome VARCHAR(100),

Indirizzo VARCHAR(255),

Telefono Varchar(20)

);

CREATE TABLE Categoria (

ID\_Categoria INT PRIMARY KEY,

Nome VARCHAR(100)

);

CREATE TABLE Recensione (

ID\_Recensione INT PRIMARY KEY,

Testo TEXT,

Voto INT,

ID\_Cliente INT,

ID\_Libro INT,

FOREIGN KEY (ID\_Cliente) REFERENCES Cliente(ID\_Cliente),  
FOREIGN KEY (ID\_Libro) REFERENCES Libro (ID\_Libro)

);

CREATE TABLE Categoria (

ID\_Libro INT,

ID\_Categoria INT,

PRIMARY KEY (ID\_Libro, ID\_Categoria),

FOREIGN KEY (ID\_Libro) REFERENCES Libro(ID\_Libro),

FOREIGN KEY (ID\_Categoria) REFERENCES Categoria(ID\_Categoria)

);