**Conceptual Design**

The most basic general, basic entities I could think of for this database are:

books

orders

publishers

users

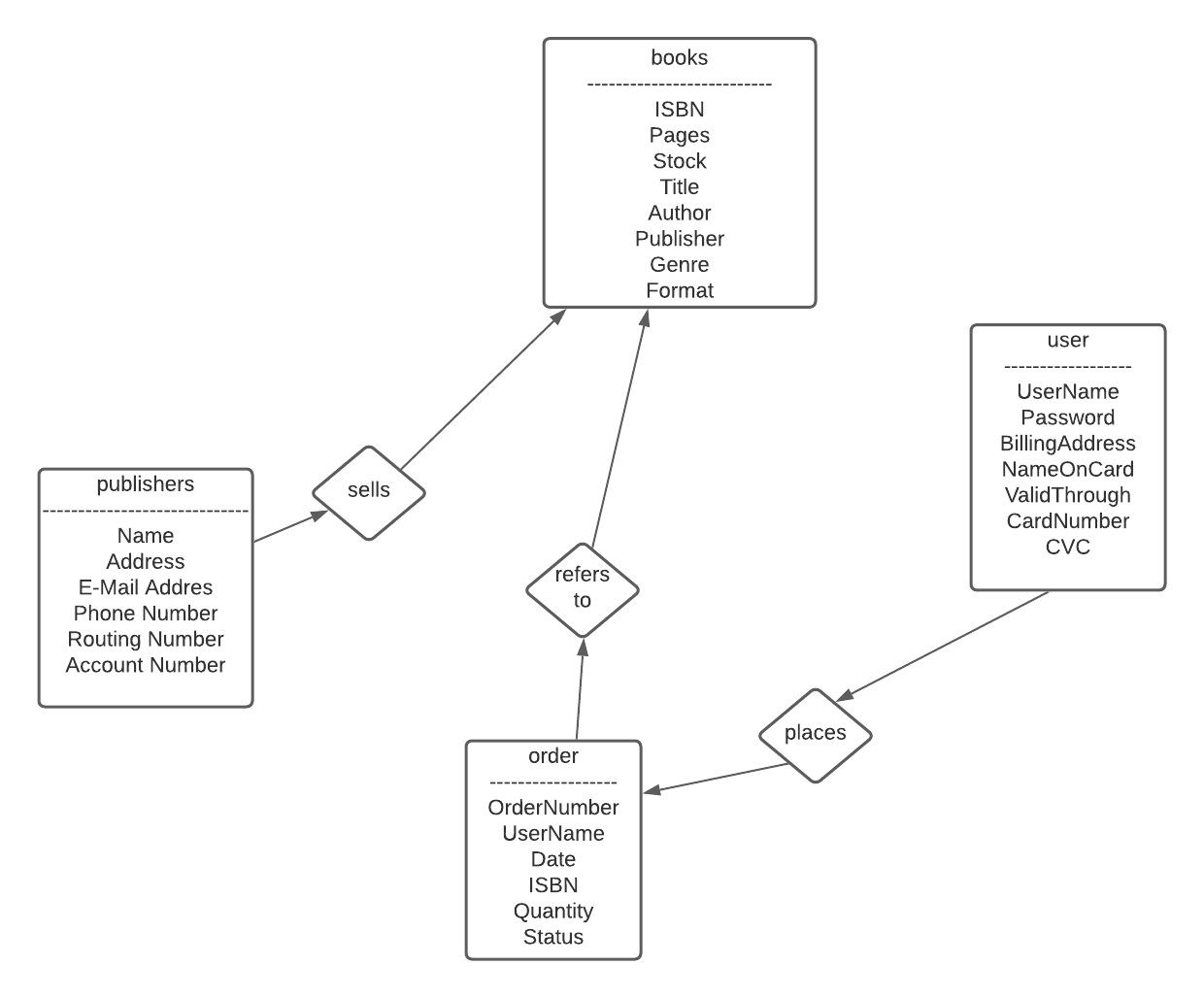
Books contain information like their unique ISBN, the number of pages the book contains, how many the database has in stock, the title of the book, the name of the author, the name of the publisher, the genre of book, and the format of the book (hardover, paperback, audiobook, etc.)

Orders contains the unique order number, the user who placed the order, when the order was placed, the ISBN of the book being purchased, how many books are being purchased, and the current state of the order (pending, in transit, shipped, etc.)

Users contain their user name, their password, and key billing information to process a transaction

The publisher has contact information, as well as basic information needed to transfer funds to them.

**ER Diagram**



This shema is not well normalized. In order to place an order for multiple books, for instance, you have to repeat the same order number in the order table for each line item in the transaction.

Reduction to Relation Schemas/Normalization

**List of functional dependencies:**

ISBN**->** (Pages, Stock, Title, Price, Format)

OrderID**->** (Date, Status)

OrderID,ISBN**->** (Quantity)

UserID**->**(UserName, Password)

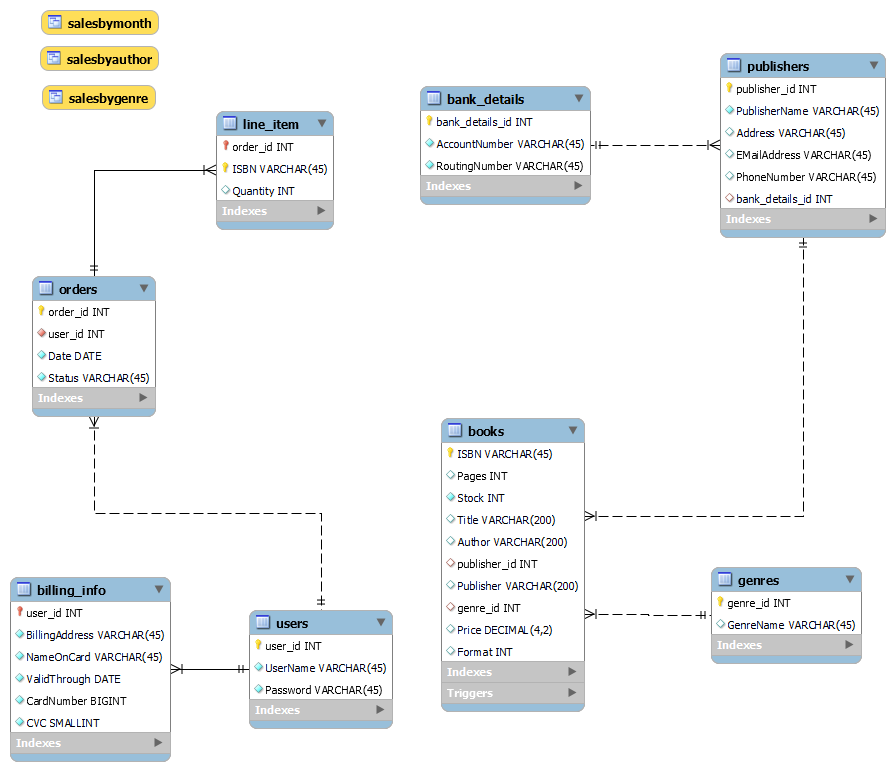
CardNumber-**->**BillingAddress, NameOnCard, ValidThrough, CVC)

PublisherID**->**(PublisherName, Address, EMailAddress, PhoneNumber)

AccountNumber**->**(RoutingNumber)

(note: my final schema diagram does not exactly follow the functional dependencies, for instance credit card number is not the primary key of billing\_info, rather it just joins on the user\_id)

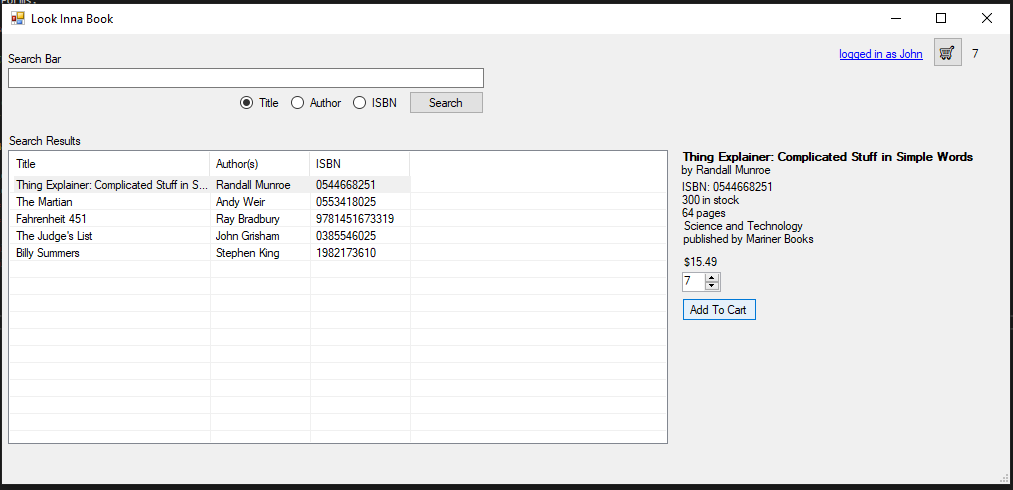
**Database Schema Diagram**

****

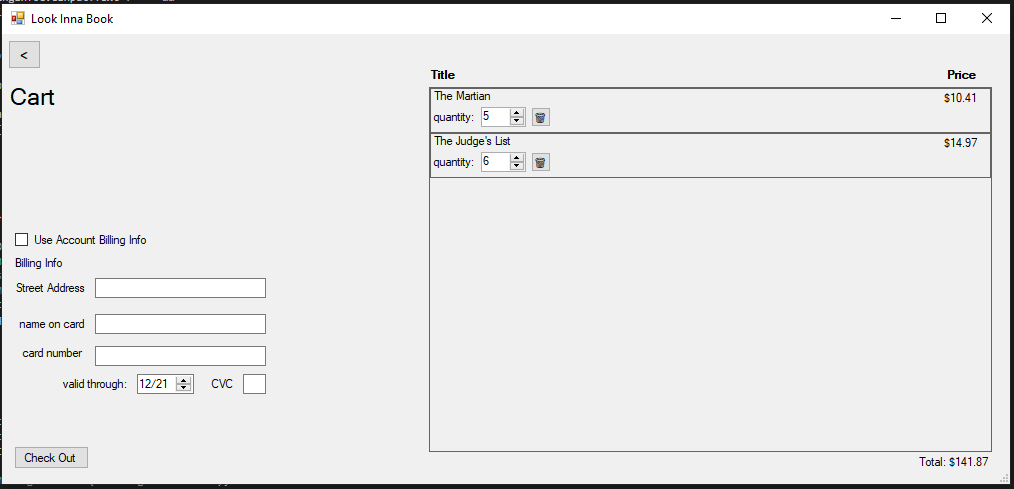
Implementation

For this project I decided to make a desktop application using C# and .NET with Windows Forms. There are two separate applications, one is a form for users to browse, login and buy books, and the other is the admin app which allows you to edit books and publishers in the database and view reports.

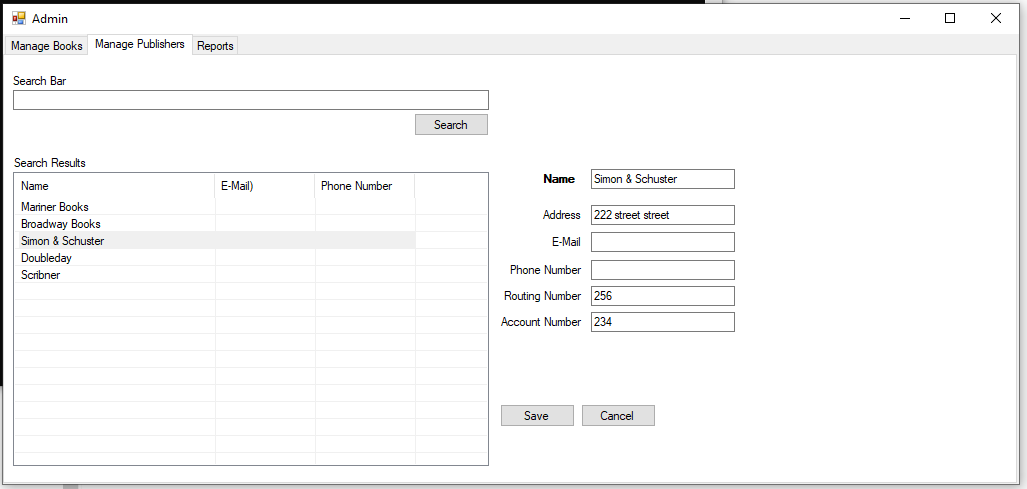
Browsing books:



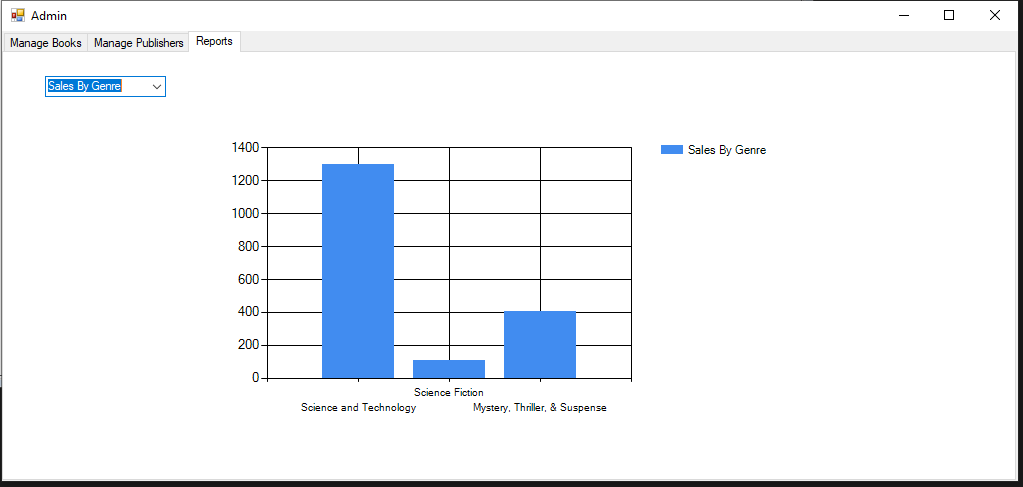
Examining your cart before checkout:



The basic interface of the admin app is similar. Here it is editing a publisher entry:



And here is the reports tab, graphing out sales by genre (reports are implemented via views into the SQL database)

****

**GitHub Repository:**

(link to github repo)  
  
**Appendix I**

availability