NotChegg

Design Document Version 1

October 30, 2017

Group Members:

Tyler Anthony

Bo Fulgham

David Gullett

Dillon Harrison

Zach Martin

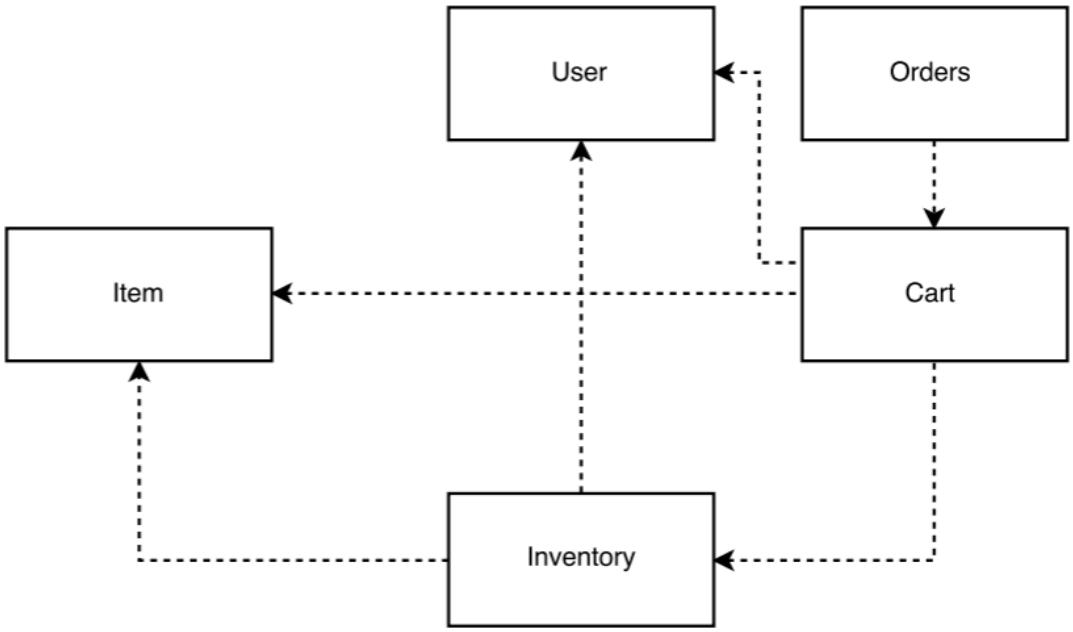
Group K

Lab Instructor:

Ben Wiggins

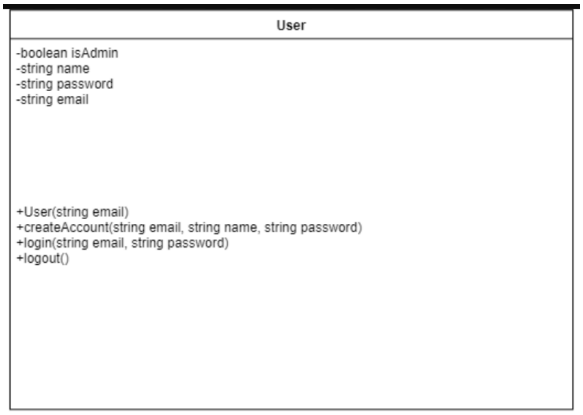
Table of Contents

1. Overview class diagram



2. Detailed class diagram

2.1.1 User



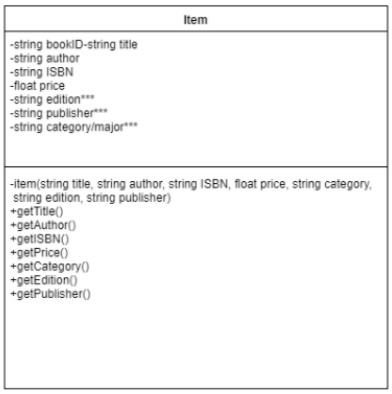
Class: User

Descriptions:
-createAccount(string email, string password string address): Constructor, assign userID and adds userID, email, password, and address to the User Table of the database.

-login(string email, string password): Query the database. If email and password combination are true return True else returns false and re-display login page.

-logout(): returns the user to login page.

2.1.2 Item



Class:Item

-Item(string title string author string ISBN string price string category string publisher): Constructor. Sets the information about the book such as title, author, ISBN, price, category, and publisher. Stores this information into the book table in the database.

-getTitle(): Query the database. Retrieve book title from database and store.

-getAuthor(): Query the database. Retrieve book author from database and store.

-getISBN(): Query the database. Retrieve book ISBN from database and store.

-getPrice(): Query the database. Retrieve book price from database and store.
-getCategory(): Query the database. Retrieve book category from database and store.
-getPublisher(): Query the database. Retrieve book publisher from database and store.

2.1.3 Inventory

Class:Inventory

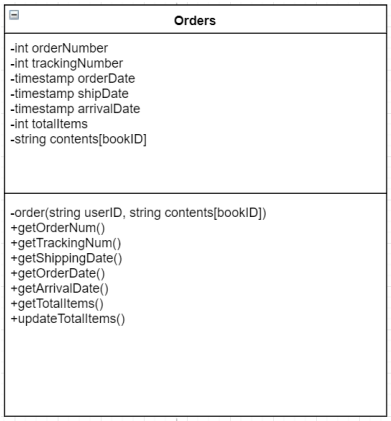
-bool add(string bookID) – creates a book object and returns a TRUE or FALSE if it succeeds or fails respectively.

-bool remove(string bookID) – deletes a book object and returns a TRUE or FALSE if it succeeds or fails respectively.

-string flag(string bookID, int reason) – Marks a book object to be reviewed for deletion by an admin from a numbered list of reasons.

-search(string searchTerms) – creates a query based on the information user input into the function.

2.1.4 Orders



Class:Orders

-order(string userID, string contents[bookID]) – creates an order with an order number, tracking number, shipping date, order date, arrival date, number of items it contains, and the contents of the order.

-getOrderNum() – returns the order number of the order

-getTrackingNum() – returns the tracking number of the order

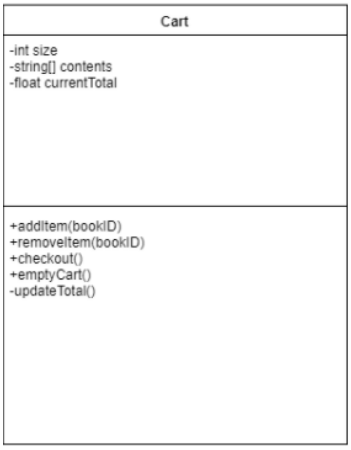
-getShippingDate() – returns the shipping date of the order

-getOrderDate() – returns the date the order was place

-getArrivalDate() – returns the date of arrival of the order

-updateTotalItems() – updates the number of items the order contains

2.1.5 Cart



-addItem(bookID) - adds the item of bookID to the string[] contents array

-removeItem(bookID – removes the item with the bookID passed into the function from the

-string[] contents array

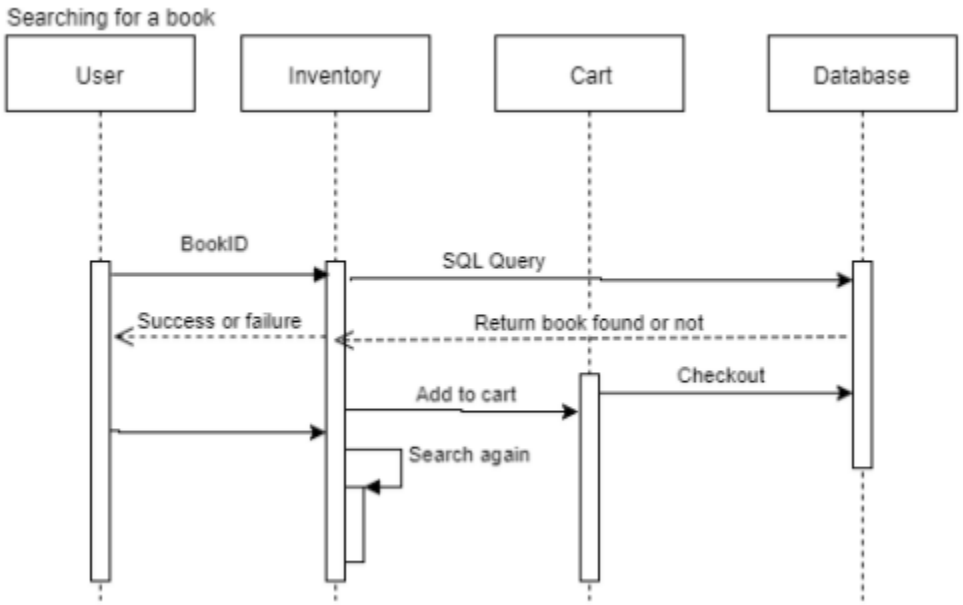
-emptyCart() – removes all bookID's from the string[] contents array

-updateTotal() – adds up all of the bookID's prices in the string[] contents array

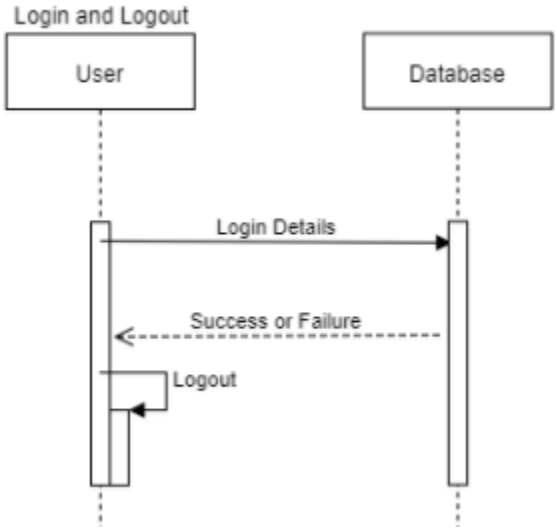
-checkout() – checkout will match user information with purchasing of the items within a cart. Checkout will add up the total price of all the books with a bookID in the contents array, and store information that is nessesary to purchase books, such as address and order numbers from the orders class

3. Sequence diagrams

3.1.1 Purchase Books



3.1.2 Logging In/Out



Appendix A. Database Design

Primary key *

Foriegnkey #

- **Users**
  *email
  Password
  Address
  isAdmin
  major

- **Inventory**
  *bookID
  Isbn
  Title
  Author
  Publisher
  Price
  Image
  isFlagged
  edition
  category

- **Orders**
  *orderNum
  TrackingNum
  orderDate
  shipDate
  arrivalDate
  ItemsOnOrder
  TotalPrice
  #bookID
  #userID

Appendix B. Task & Role assignments

Tyler Anthony – User interface

Zack martin – Database

Dillon Harrison – Authentication/Database

Bo Fulgham – Database Queries

David Gullet – General Programing