Project Name: TigreNoble

Team: Kemoy Campbell, Kabo Cheung, Tom Ansill

Problem:

Up to now RIT students has been using Facebook pages called “Free and For Sale RIT and beyond” and “Free & For Sale” to sell their belongings on sale like used books, laptops, furniture, bicycles, etc. In most cases, searching for books can be time consuming as they have to scroll on forever to find a particular book. In addition, sometime a book have been sold but has not deleted from the page. Hence, someone could have thought that the book is still for sale, and he or she will only end up disappointed to find out that the book has been sold 3 months ago.

Proposal:

We will design a convenient system where students could sell books, and the system will keep the inventory up to the date. The system will store the book information, such as author name, description, edition, etc, by requesting the book ISBN number from the user and use google API to fetch the necessary data, such as author name, description, edition, etc. It will also store the name of students and the relationship, such as seller and buyer. The seller can accept or decline the buyer’s offer. Once accepted, both parties can discuss where and when to deliver through our system messaging. We will implement a bidding system, which is optional to the seller and buyer. Once a book has been sold, its post information will automatically be deleted from the system.

System/Application:

We will use PHP, MySQL, Apache, and RIT’s LDAP server. With RIT’s LDAP server, we can authenticate DCE accounts and passwords while not storing passwords on our server. It also retrieves information about students which we will use it to automatically register them if they are not already in the system.

Features:

Student

DCE, name, local address, email

Book

isbn, year_written, title, author, publisher, edition, description

Authors: Kemoy Campbell, Kabo Cheung, Thomas Ansill

Messaging

Inbox

id, sender_dce, receiver_dce ,dateReceived, subject, message

Post

id, seller_dce, book_isbn, book_condition, book_price, post_date, book_description (have code, tear, highlight etc), post_type (sell,trade, and both), post_status(sold, pending transaction), book_quantity, imagePath

Bidding

Bid_money

posting_id, student_dce, bid_amount, bid_accept

Bid_trade

posting_id, student_dce, offer_message, trade_accept

Rating

reviewee, reviewer, rating, message

Authors: Kemoy Campbell, Kabo Cheung, Thomas Ansill

Data Domain

RIT's LDAP Server:

RIT'S lDAP server will be use to verify if users' RIT DCE and password are valid. Thus removing the need for us to store the user password on our system. This also gives the user one less account to worry about. The basic information of the student such as full name, dce and email will be fetch from LDAP if they are a first time user and store in the "student" table.

Our Database:

*Note: Primary key

Student: The student table will be used to store all information about a user. The table stores information such as user's DCE as a primary key, user's full name (name will be received from RIT's LDAP server), email (received from RIT's LDAP server), and user's local address.

Table Column Attributes

dce (varchar): primary key - user's username

name (varchar): user's fullname

email (varchar): user's email address

local address (text): user's local address so system can share the address to other customer/seller after a sale have been made. it is student local address.

Post: The posting table will be used to store all information about book sale postings.

Table Column Attributes

id (int): Auto-incrementing number of each postings for easy reference

seller_dce (varchar): the seller's username

book_isbn(bigint): unique book number used to identify which book

book_condition (varchar):indicates whether if book is new, good, or poor

post_status (char): indicates whether if book has been available or sold

post_type(char): indicates whether if the post is for bidding or trading

book_isbn (bigint): reference to book ISBN table

post_date (date): shows how old the post is. We will use that to remove any posts older than 30 days so the database stays clean.

book_condition (varchar): describes the condition of the book

book_price (decimal): suggested price for the book

book_quantity (int): quantity for the book

book_description (text): users' description of the book and its condition, etc.

imagePath (text): URL path to the specific image for the book

Authors: Kemoy Campbell, Kabo Cheung, Thomas Ansill

Book: Local storage of book information. When a new book is introduced, we will download all necessary information from google's books API and store it to the table.

Table Column Attributes

isbn (bigInt): unique book number used to identify which book
title (varchar): title of the book
author (text): author of the book
publisher (varchar): publisher of the book
edition (smallInt): edition of the book if applicable
description (text): official description of the book

Rating: Table used for users' ratings so users can rate their sellers' performance, trustworthy, and quality of service.

Table Column Attributes

reviewee (varchar): identity of the user who is getting the review
reviewer (varchar): identity of user who is making the reviews
rating (int): a rating number between 1 to 5
message (text): review message

Inbox: Table used to facilitate the communicate between two users across the site.

Table Column Attributes

id (int): id of the message
receiver_dce (varchar): user who receives the message
sender_dce (varchar): user who sends the message
dateReceived (datetime): date of the message received
subject (varchar): subject of the message
message (text): message body

bid_trade: Table used to store non-monetary posting offer.

Table Column Attribute

id (int): id of user
posting_id (int): id of post that the offer was referred to
student_dce (varchar): id of RIT student user
offer_message (text): message of the offer
trade_accept(char): confirmation that determines if the offer is accepted

bid_money: Table used to store monetary posting offer

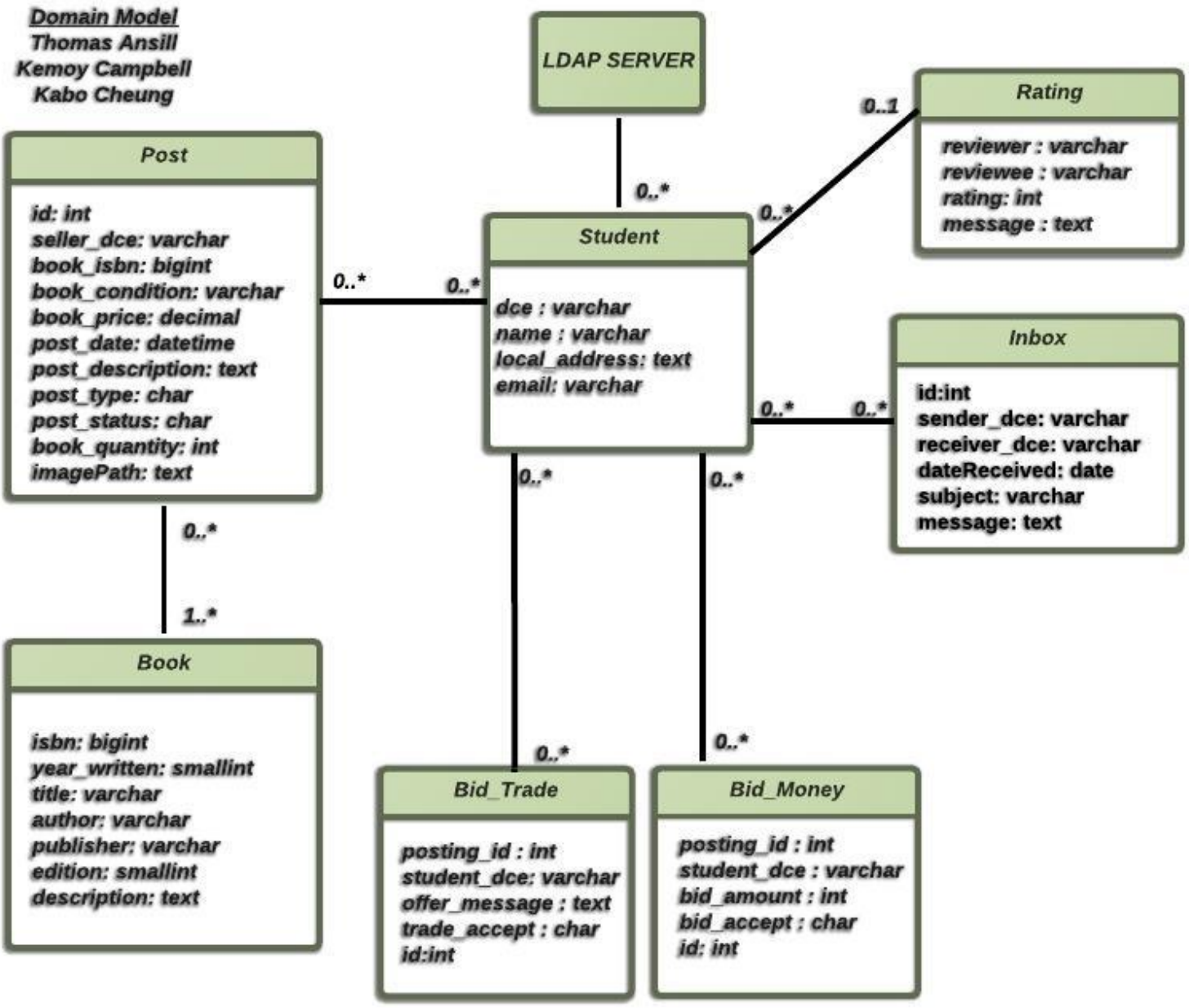
Table Column Attribute

id (int) - id of user
posting_id (int): id of post that the offer was referred to
student_id (varchar): id of RIT student user
bid_amount: the amount of bid from buyer
bid_accept (tinyint): confirmation that determines if the offer is accepted

Authors: Kemoy Campbell, Kabo Cheung, Thomas Ansill

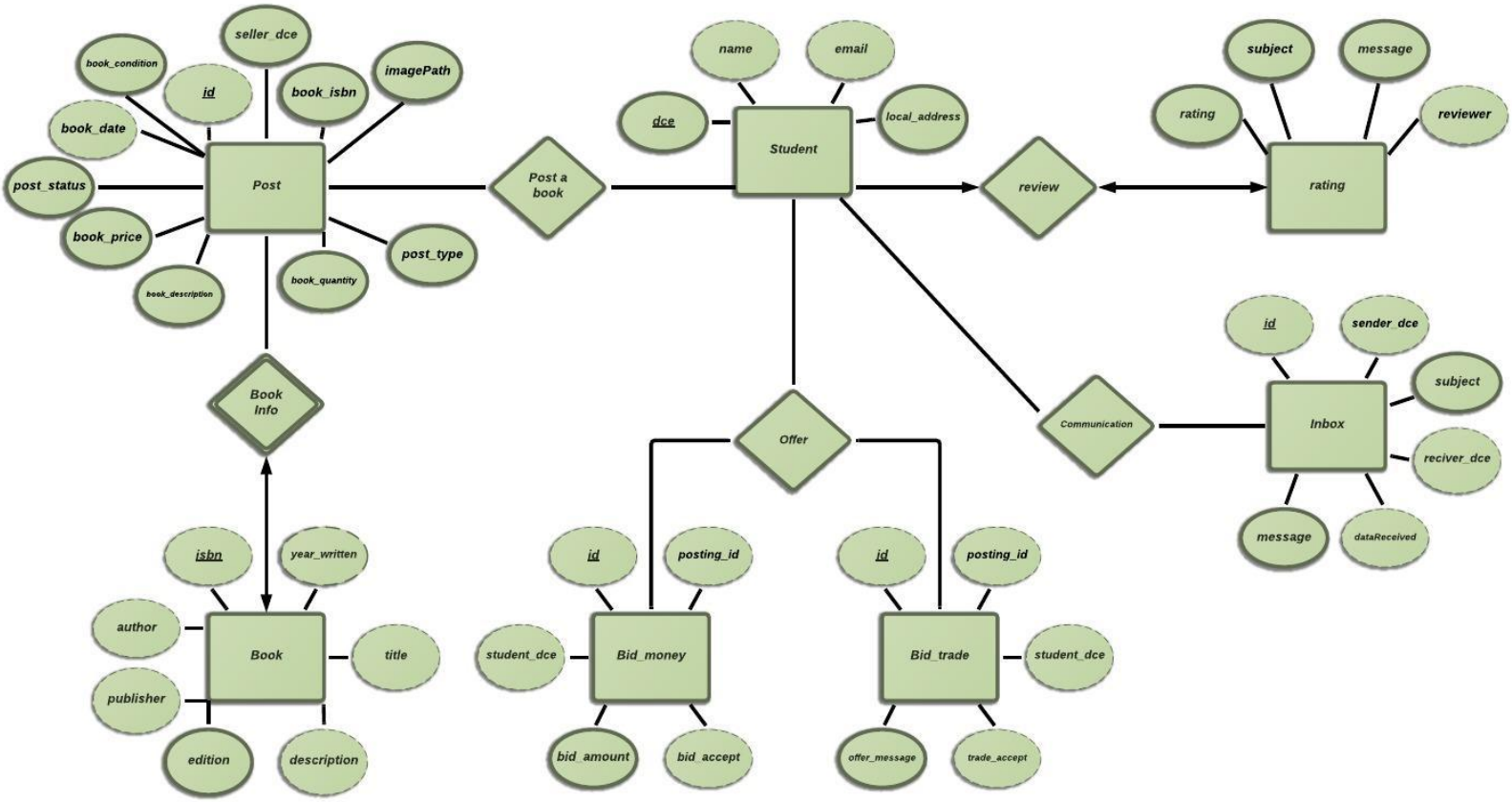
Authors: Kemoy Campbell, Kabo Cheung, Thomas Ansill

Data Domain Model



Authors: Kemoy Campbell, Kabo Cheung, Thomas Ansill

ER Diagram



Summary/Goal

To design a convenient system where students could sell books, and the system will keep the inventory up to the date. The system will store the book information such as author name, description, edition, etc, by requesting the book ISBN number from the user and use google API to fetch the necessary data, such as author name, description, edition, etc. It will also store the name of students and the relationship, such as seller and buyer. The seller can accept or decline the buyer's offer. Once accepted, both parties can discuss where and when to deliver through our system messaging. We will implement a bidding system, which is optional to the seller and buyer. Once a book has been sold, its post information will automatically be deleted from the system.