

# DESIGN DOCUMENT: BOOKS DATABASE

Ashish Gupta  
Y8140

Vishal  
Y8578

Group No.09

## ABSTRACT

The project is based on a book database system pertaining to various needs of the user. The basic interface involves querying books according to language, title, author, publisher, ISBN. We support services for buying and selling used books or books used in specific IIT Kanpur courses. We build a personal profile page which is used for handling the transactions between various students. We implement a "recommendation system" for recommending books to be used in a particular course in addition to their availability in the library. The system gives advice for cheap used books available at the time, when that book is not found in the library.

## OUTLINE

1. Information regarding the book: Set of entities that support the store of books, authors and publication and general search queries for their availability in the library. Advanced search queries are also provided.
2. Information regarding buying/selling: Set of entities that support personal profile of the students and transactions. It also handles notifications to be sent to the sellers as well as the buyer when there is a successful transaction.
3. Information regarding courses at IITK: Stores recommended books for different courses at IITK, which in turn are searched directly in library and results are provided to user.

## QUERIES AND FUNCTIONALITY

1. The students can browse the book without logging in, however in order to buy/sell books they need to log in with a username and password and the username can be used as key for students.
2. A login id for a particular student is created when the student registers himself at the portal. The system checks for any existing user with same desired userid. If found, the system gives direction to user to choose another username.
3. On logging in, the system moves to personal profile page which will show notifications regarding the current books sold and bought by the user. The page will hold personal information and various options available to the user.
4. Various queries that can be taken by the user:
  - (a) **General search for a book:**
    - The user can search for books in the library section or in the buy-sell section.
    - Accordingly the LIBRARY table or the SELLER table would be used for the search.

- Options such as language, title, author, publisher, ISBN, publication date, used or new, price are available.
- We provide features of a partial search of a phrase or exact words in the phrase.

(b) **Add a book to the database:**

- The person specifies complete information regarding the book along with the information about the authors and publication details.
- If the book is not present in the BOOK table, the BOOK table is updated with the entry.
- If the authors of the particular book are not present, the corresponding entries are added.

(c) **Selling a book:**

- A drop down menu indicating books that can be sold are provided to the user.
- If the concerned book is not present in the drop-down menu,
- The person specifies complete information regarding the book along with the quantity and price.
- An entry is added to SELLER table specifying the seller id.

(d) **Buying a book:**

**We provide a quick purchase strategy in the search itself.**

- As soon the person clicks on the buy button, a webpage requiring information for payment options opens.
- The count of that item sold by the user is deducted and if no item remains, its entry is removed from the SELLER table.
- A record of the transaction is added to TRANSACTION table, which handles the credit information regarding the transaction and BUYER-SELLER table, which holds the relations between buyer and seller is also updated.
- An entry is also added to the NOTIFICATION table to be displayed when the buyer and seller next time logs in.

(e) **Storing Search Results:**

- We provide the option of user returning to a particular search result which it has last carried out in the middle of any transaction.
- This feature was implemented using **session variables**.

(f) **Transaction history:**

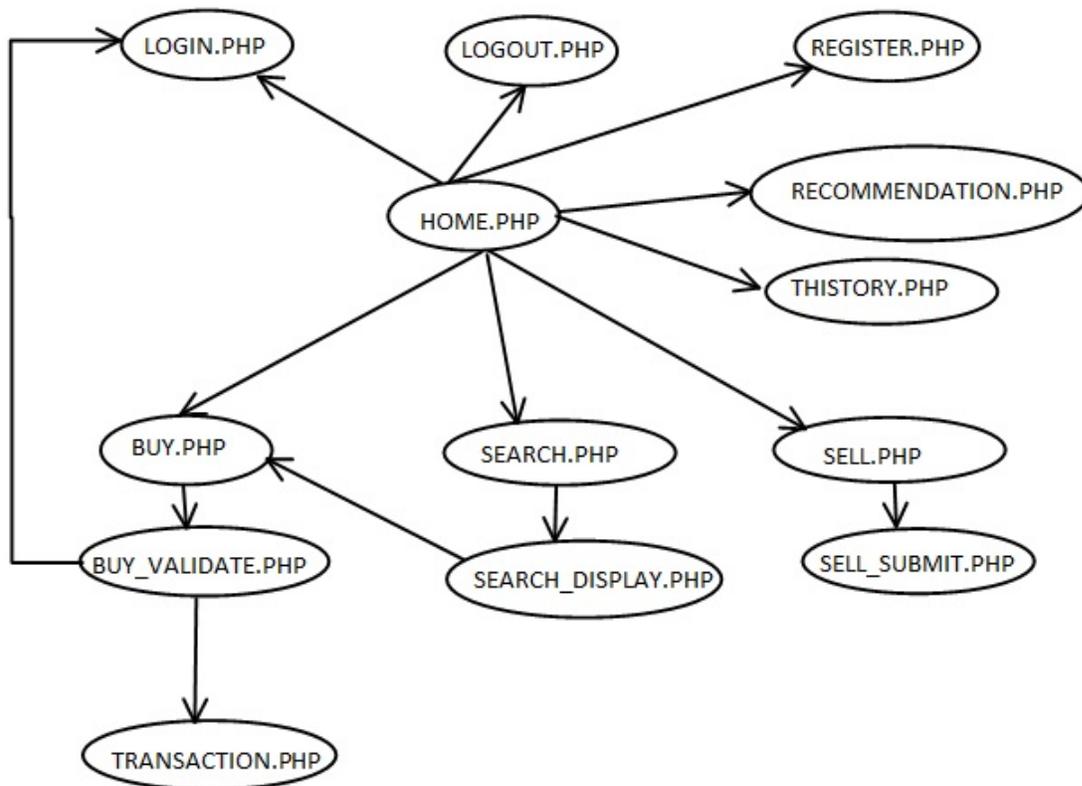
- Any student whether a buyer or seller, can view his all transactions up to date.
- We traverse the BUYER-SELLER table and search for all the entries in which the user id appears in the buyer or seller table and sort them accordingly.

(g) **Notification:**

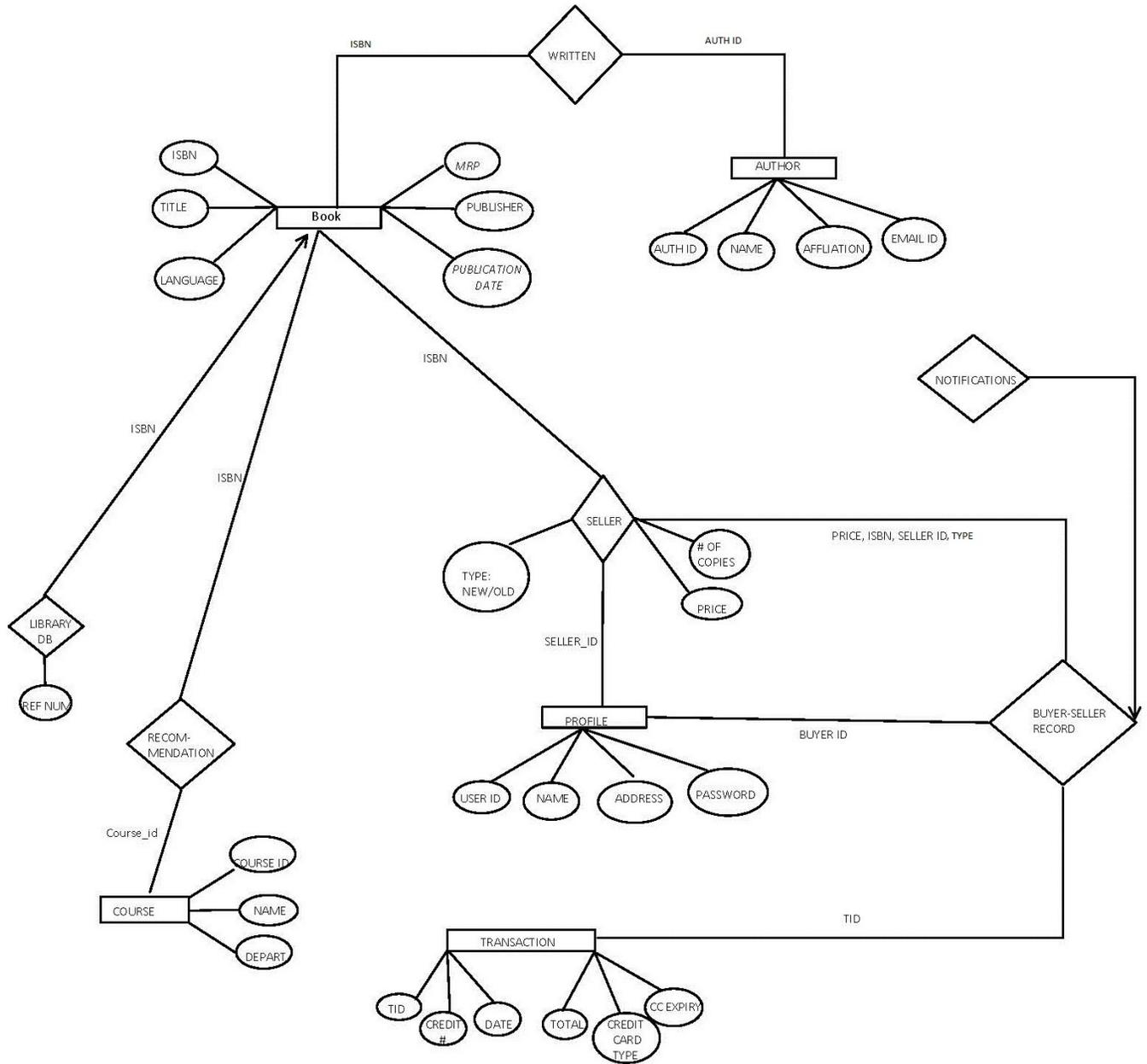
- Any time a transaction is carried out, besides entering the record in the TRANSACTION and the BUYER-SELLER table, it is also entered in the NOTIFICATION table.
- Whenever the seller or buyer logs in, its corresponding entries are fetched from the NOTIFICATION table. The user is supplied with the option to remove the notifications upto the point that has been seen by the user.

(h) **Course Books available:**

- The student selects the course number which displays the recommended books for that course.
- The system gives recommendation about the book if it's available in the library or if it is being sold by someone.
- The course is searched in the COURSE table. There is a one to many relation between course and book.
- For each book corresponding the course, its ISBN number is searched in the LIBRARY table or the SELLER table.



FLOW CHART: Books Database



E-R Diagram: Books Database

## TABLES

1. **Book**(ISBN, title, language, MRP, publisher, pub\_date)
2. **Author**(auth\_id, name, affiliation, email)
3. **Profile**( user\_id, user\_name, address, password)
4. **Transaction**( tid, date, total, creditcardNum, cc\_type, cc\_expiry)
5. **Course**( course\_id, course\_name, department)
6. **Written**(ISBN, auth\_id)
7. **Library**(ref\_num,ISBN)
8. **Recommendation**(ISBN, course\_id)
9. **Seller**(ISBN, seller\_id, type, num\_copies, price)
10. **BuySellRecord**( seller\_id, buyer\_id, stype, ISBN, tid)
11. **Notification**(seller\_id, buyer\_id, ISBN, tid)

## Non-Trivial Functional Dependencies

1. Book:-  
ISBN  $\rightarrow$  (title, language, MRP, publisher, pub\_date)
2. Author:-  
auth\_id  $\rightarrow$  (name, affiliation, email)
3. Profile:-  
user\_id  $\rightarrow$  (user\_name, address, password)
4. Transaction:-  
tid  $\rightarrow$  (date, total, creditcardNum, cc\_type, cc\_expiry)
5. Course:-  
course\_id  $\rightarrow$  (course\_name, department)
6. Written:- No functional dependency
7. Library:-  
ref\_num  $\rightarrow$  (ISBN)
8. Recommendation:- No functional dependency
9. Seller:-  
(ISBN, seller\_id, type)  $\rightarrow$  (type, num\_copies, price)
10. BuySellRecord:- No Functional dependency
11. Notification:- No Functional dependency

## Normalization

Book:-

ISBN  $\rightarrow$  (title, language, MRP, publisher, pub\_date)

Note that the ISBN is the candidate key for BOOK table. The corresponding set F+ corresponding to the above functional dependency can be divided into two sets ( rule  $\alpha \rightarrow \beta$ ):

- 1)  $\alpha$  contains the candidate key, ISBN, then  $\beta$  would contain the whole schema attributes.
- 2)  $\alpha$  does not contain key, ISBN, then  $\beta = \alpha$

Now we will check the BCNF condition for each set.

For set-1,  $\alpha$  is the superkey key and hence satisfies the condition.

For set-2,  $\alpha \rightarrow \beta$  is a trivial functional dependency.

Hence, we can see that if the non-trivial functional dependency contains the candidate key only, then the resulting schema would be in BCNF.

The SELLER table is a unique case in which the three tupe together consitutes the primary key. To ensure BCNF, we need to make sure that no other attribute from the seller table appears along with this three tuple in some other table. The given observation holds in our database and hence SELLER table is in BCNF form.

Relations with no non-functional dependencies: It can be clearly seen that only trvial functional dependencies applies to the given schema and hence they are already in BCNF.

Hence all indivicidual schemas are in BCNF, which implies the whole database is in **BCNF**.

## Implementation

The project was coded in PHP 5 with database implementation in Mysql. The validation of every form is implemented at the client side using JavaScript. Session variables have been used to maintain data during navigation between different pages. For example, a session variable corresponding to username is set when the user logs in. In case of buy feature, this session variable is checked to be set. If not, the system gives the option to login or register at the same page. Search results are also maintained during a transaction using session variables. The implementation detail of particular queries is mentioned in the functionality section above.

## Snapshots

**BOOK MANAGEMENT SYSTEM**

**LOGIN TO ENTER**

USER ID

PASSWORD

**Do not have an id, [Register](#)**

**GENERAL USER SEARCH**

OPTIONS SEARCH  BUY

ISBN

TITLE

LANGUAGE

PUBLISHER

AUTHOR NAME

LIB REFERENCE NUM

Home: Books Database

SELECT ISBN :

Selling Price

Number of Copies

Book Type: NEW  USED

Sell: Books Database



SELECT Course : 'CS101'

### Displaying Recommended Books for Course : CS350

ISBN	TITLE	LANGUAGE	MRP	
234-564-23	Compiler Design	English	400.00	<input type="button" value="Search/Buy"/>
34-432-22	Machine Learning and Pater	English	700.00	<input type="button" value="Search/Buy"/>

Recommend: Books Database

### Displaying all possible sellers of the selected book for isbn :43-235-987

TITLE	SELLER ID	SELLER ADDRESS	BOOK TYPE	NUM COPIES	PRICE	
Database Systems Concept	lennon	30 Rockefeller center, NY	new	13	400.00	<input type="button" value="Confirm"/>
Database Systems Concept	lennon	30 Rockefeller center, NY	old	5	200.00	<input type="button" value="Confirm"/>
Database Systems Concept	satch	The Chickenfoot plaza, Stor	new	12	390.00	<input type="button" value="Confirm"/>
Database Systems Concept	satch	The Chickenfoot plaza, Stor	old	7	190.00	<input type="button" value="Confirm"/>
Database Systems Concept	hammett	Metal woods avenue	new	24	380.00	<input type="button" value="Confirm"/>
Database Systems Concept	hammett	Metal woods avenue	old	14	200.00	<input type="button" value="Confirm"/>
Database Systems Concept	morello	30, Nehru place, New Delhi	new	13	390.00	<input type="button" value="Confirm"/>
Database Systems Concept	morello	30, Nehru place, New Delhi	old	6	250.00	<input type="button" value="Confirm"/>
Database Systems Concept	vai	Chandani chowk, Stall Num1	new	5	385.00	<input type="button" value="Confirm"/>
Database Systems Concept	vai	Chandani chowk, Stall Num1	old	8	205.00	<input type="button" value="Confirm"/>

Seller: Books Database

#### Transaction Details

BUYER ID

SELLER ID

ISBN

BOOK TYPE

PRICE

DATE

CREDIT CARD NUMBER

CREDIT CARD TYPE

CREDIT CARD EXPIRY

Transaction: Books Database