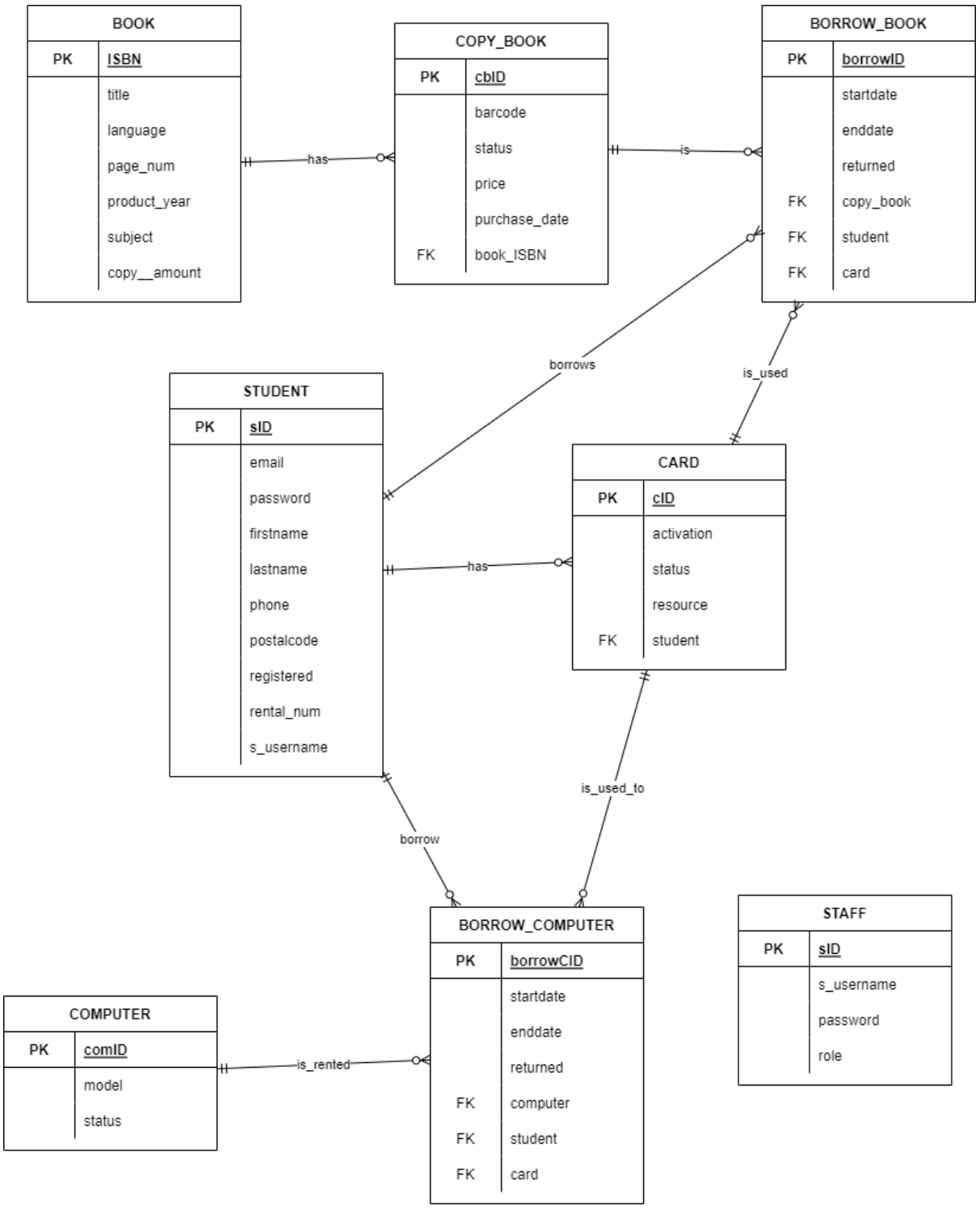
Projectreport

by Dmytro Shutenko, Wei-Chieh Wang, Nathalie Juliette Schlech

Conceptual Data Model



We designed the CDM Diagram with a tool offered by Visual Studios. We decided to use that instead of the other tools, because it supports the Crow Foot Notation and it is delivered in English, so its easy for all of us to understand the tool.

We have 8 tables. In the following we explain some of the tables and some of their attributes. The table 'staff' combines the administrator and the librarian. The attribute role shows if the user is a 'librarian' or an 'administrator'. The table 'student' consists of all the information about the students. The 'registered' attribute gives information about if a student is registered or not. Depending on that information a student can borrow a different number of resources at the same time. The attribute 'rental_num' contains the number of resources, that are borrowed by the student at the moment. The table 'card' contains information of the cards. Each card is assigned to one student, and each card can only be used for one specific resource. The attribute 'resource' gives information for which resource type the card can be used. The attribute 'activation' contains the activation date of the card and 'status' contains if a card is activated or deactivated at the moment.

The table 'book' has all information about the different books. In the column 'copy_amount' is the number of copies saved, that the library owns. Each book has 0 to n copies. These copies are saved in the table 'copy_book'. The attribute 'book_ISBN' is the foreign key, that declares which book the copy is assigned to. The attribute 'status' describes if the copy is borrowed at the moment or if it is available. Other than books our system also manages the resource computer. The table 'computer' has the attribute 'status' too. It describes if the computer is borrowed at the moment or if it is available.

The last two tables manage the borrow relationship. 'borrow_book' and 'borrow_computer' store information about who borrowed what with which card and how long are they borrowing it. The attribute 'returned' saves if the resource is still borrowed or already returned.

Our way through the project

We decided to use Visual Studio for the CDM Diagram. After that we created the Database with all 8 tables with MySQL and created all triggers and Constraints at that time. At the same time, we started to work on the graphical interface. We decided to use phpAdmin for this because it works with MySQL. We designed the frontend interface and created different Logins for the different user types and connected it with the database. We also created csv. -files with the data to populate our database tables. The create_table.sql and the csv. -files are attached in the folder we handed in. The following table shows the different stages more detailed and the time we spend working on each point:

	Stage	**Hours spent**
	Reading requirements	1
	Idea formulation	1
	Environment set up: WAMP local server	0.5

	Idea implementation research (approaches, best practices)	2
	Database design (CDM)	2
	Structural constraints of the database listing	1
	A script to create the tables (SQL)	2
	SQL queries to interact with DB	2
	DB population	1
	GUI development / Front-end: (HTML, CSS, JS)	10
	Back-end (PHP, MySQL)	15
	Testing	5
	Reporting	1.5
	Presentation Preparation	1
Total:	**14 stages**	**45 hours**

List of structural constraints

1. Staff
   - PRIMARY KEY: sID
   - UNIQUE: s_username
   - CHECK: role IN ('admin', 'librarian')
   - AUTO_INCREMENT: sID
   - NOT NULL: s_username, sID, password

2. Student
   - PRIMARY KEY: sID
   - UNIQUE: s_username, email
   - CHECK:
     - registered IN (0,1)
     - email` LIKE '^[(a-z)][(a-z)|(0-9)]*@[a-z]+[.][a-z]{2,3}$'
   - AUTO_INCREMENT: sID
   - NOT NULL: sID, s_ username, password
   - DEFAULT:
     - firstname (NULL)
     - lastname (NULL)
     - email (NULL)
     - phone (NULL)
     - postal_code (NULL)
     - registered (0)
     - rental_num (0)

3. Card
   - PRIMARY KEY: cID
   - CHECK:
     - `status` IN ('active', 'inactive')
     - `resource` IN ('book', 'computer')
   - AUTO_INCREMENT: cID

- NOT NULL: cID
- DEFAULT:
    - activation (NULL)
    - status ('inactive')
    - resource (NULL)
    - student (NULL)
- FOREIGN KEY: student REFERENCES student (sID)

4. Book
    - PRIMARY KEY: ISBN
    - CHECK:
        - page_num > 0
    - NOT NULL: ISBN
    - DEFAULT:
        - language (NULL)
        - title (NULL)
        - page_num (NULL)
        - product_year (NULL)
        - subject (NULL)
        - author (NULL)
        - publisher (NULL)
        - copy_amount (0)

5. Copy_book
    - PRIMARY KEY: cbID
    - CHECK:
        - `price` > 0.0
        - status IN ('borrow', 'available', 'unavailable')
    - AUTO_INCREMENT: cbID
    - NOT NULL: cbID, book_ISBN
    - DEFAULT:
        - barcode (NULL)
        - price (NULL)
        - purchase_date (NULL)
        - status ('unavailable')
    - FOREIGN KEY: book_ISBN REFERENCES book (ISBN)

6. Computer
    - PRIMARY KEY: comID
    - CHECK:
        - `status` IN ('borrow', 'available', 'unavailable')
    - AUTO_INCREMENT: comID
    - NOT NULL: comID
    - DEFAULT:
        - model (NULL)
        - status ('unavailable')

7. Borrow_book

- PRIMARY KEY: borrowID
- CHECK: `returned` IN (0,1)
- AUTO_INCREMENT: borrowID
- NOT NULL: borrowID, student, card, copy_book
- DEFAULT:
    - startdate (NULL)
    - enddate (NULL)
    - returned (0)
- FOREIGN KEY:
    - Card REFERENCES card (cID)
    - Copy_book REFERENCES copy_book (cbID)
    - Student REFERNCES student (sID)

8. Borrow_computer
    - PRIMARY KEY: borrowCID
    - CHECK: `returned` IN (0,1)
    - AUTO_INCREMENT: borrowCID
    - NOT NULL: borrowCID, student, card, computer
    - DEFAULT:
        - startdate (NULL)
        - enddate (NULL)
        - returned (0)
    - FOREIGN KEY:
        - Card REFERENCES card (cID)
        - Computer REFERENCES computer (comID)
        - Student REFERNCES student (sID)