**INF1343 - Assignment 1 Database Proposal & Design**

**Instructor: Prof. Arik Senderovich**

**Submitted by**

**Seida Ahmed**

**Misia Robins**

**Application Domain: Library Circulation**

For this assignment we have selected library circulation as an application domain.Library circulation databases help maintain library items (books and tech items), maintain and keep track of item checkouts(borrowing) and reservation. Moreover, via the database members(patrons) subscriptions and profiles will be maintained. It is also easy to locate each item, especially books based on the rack number record in the database. On top of that circulation statistics and reporting can be easily generated using the database. Based on the above mentioned advantages and other more , having a database for library circulation is highly significant.

This is a subsystem of a larger library system. For the purposes of the assignment, we are imagining the catalogue and circulation system as two different entities, linked by an entity representing the particular holding. While the catalogue deals with metadata specific to a work that the library owns at least one copy of, the circulation system tracks each copy of each work as its own object. These items are physically located at different library branches, and can be borrowed or reserved by patrons. The circulation system tracks where library items are, who has placed holds on them, and whether a patron owes any fees.

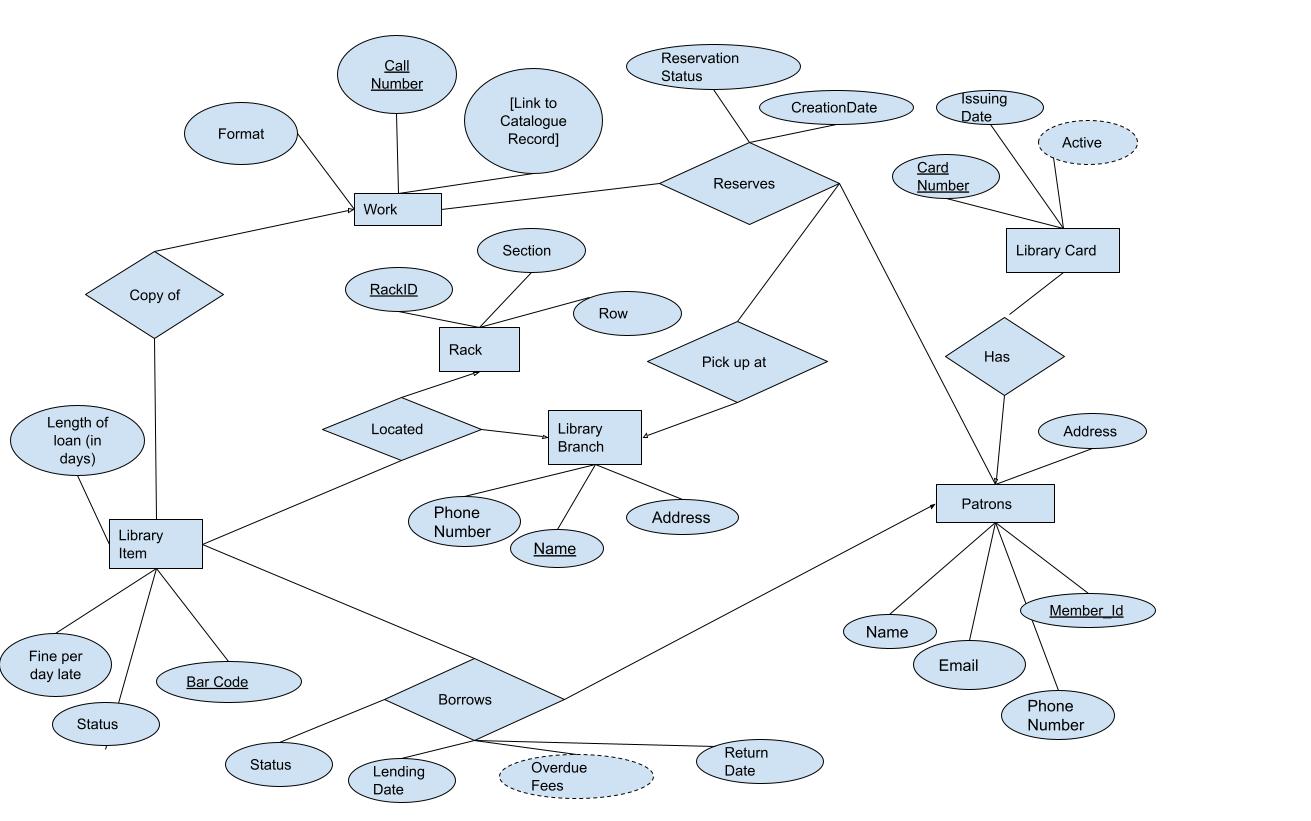
**Sample Queries**

* Is a given item available in the library at a given moment? If so, in which branch(es) is it available?
* What material is the most frequently borrowed item *and from which* branch?
* When was the borrowing date of a given item, who borrowed it currently ,determine its due date and overdue fines for a borrowed item.
* Which section of a library branch receives the most hold requests?
* In which rack a given book specifically found in a given branch?
* How many counts of circulation transactions at a specific branch in a specific timeframe?
* Are there any available copies of a reserved work? If not, when is one next due back at the library?
* Change the library card status to ‘inactive’ for patrons owing more than $50 in fines.
* Query whether a user has an active library card
* Update the fines accrued for a borrowed item

**Data Requirements**

* A ‘library item’ is an instance of a ‘work’ in the library’s catalogue: for example, the fifth edition of JRR Tolkein’s ‘The Hobbit’ is a work, and each of the library’s three copies of this edition is a separate library item.
* The ‘work’ is linked to the library catalogue. It is identified by its call number.
* For a library item, we want to know what branch it belongs to, and where in that branch it is located (we are calling this location ‘rack id’). We also want to know the item’s ‘status’ (is it missing, damaged, or circulating). An item’s loan period and late fee rate will be determined based on the item’s format (coming from the catalogue record), and its status. It should also be possible to override these default values.
* For each library branch, we want to store the name and address
* Within a branch, an item is located in a specific rack of a specific section.
* For “Patron” we want to record their name, address, and some other contact information (email and/or phone).
* Each patrons have library card, library card entity records issuing date, card number and weather a given card is active or not
* A patron can *reserve* a work, and specify which branch they’d like the hold delivered to. We record the date the reservation is made. When a copy of the work is available, that item will be moved to the specified branch. At this point, a new entry will be added to the *‘borrows*’ relation.
* *Borrows* is the relation between a patron and a specific library item. This includes items that have been placed on hold after a reservation request. The borrowing transaction has a status: 1 - in transit, 2 - ready for pick-up, 3 - checked out, and 4 - returned. We also keep track of the date a hold is ready for pick up, and the date an item is borrowed. If a hold is not picked up in x amount of time, it can be put back in the main circulating collection. For borrowed items, late fees can be calculated based on that item’s loan period and the borrowing date.
* A user can have maximum *n* items borrowed at any one time.
* A borrower’s history should not be kept after they’ve returned an item and paid all late fees associated with that item. However, we still want to be able to see that the item has circulated.
* A user has one active card, but they may have prior inactive cards

**ERD Diagram**



(See larger editable version of diagram here: <https://docs.google.com/drawings/d/1H9pvrtqkVCprIzEZprHaVH0s_d_00IPNZ59keCdhzsY/edit?usp=sharing>)

**DDL (exported from MySQL Workbench)**

CREATE TABLE `borrow` (

`idBorrow` int NOT NULL AUTO\_INCREMENT,

`Item` int NOT NULL,

`Card` int NOT NULL,

`status` tinyint NOT NULL DEFAULT '0',

`date\_out` date DEFAULT NULL,

`return\_date` date DEFAULT NULL,

`fines` decimal(5,2) DEFAULT NULL,

PRIMARY KEY (`idBorrow`),

KEY `item\_borrowed\_idx` (`Item`),

KEY `borrowed\_by\_idx` (`Card`),

CONSTRAINT `borrowed\_by` FOREIGN KEY (`Card`) REFERENCES `patron` (`member\_id`),

CONSTRAINT `item\_borrowed` FOREIGN KEY (`Item`) REFERENCES `libraryitem` (`BarCode`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `branch` (

`name` varchar(45) NOT NULL,

`address` varchar(45) NOT NULL,

`phone\_num` int NOT NULL,

PRIMARY KEY (`name`),

UNIQUE KEY `name\_UNIQUE` (`name`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `library\_card` (

`card\_num` int NOT NULL,

`date\_issued` date NOT NULL,

`Active` tinyint DEFAULT '0',

`member` int DEFAULT NULL,

PRIMARY KEY (`card\_num`),

KEY `member\_idx` (`member`),

CONSTRAINT `member` FOREIGN KEY (`member`) REFERENCES `patron` (`member\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `libraryitem` (

`BarCode` int NOT NULL,

`CallNum` varchar(20) NOT NULL,

`Status` varchar(10) DEFAULT 'AVAILABLE',

`fine\_per\_day` decimal(3,2) DEFAULT '0.15',

`loan\_period` int DEFAULT '21',

PRIMARY KEY (`BarCode`),

KEY `CallNum\_idx` (`CallNum`),

CONSTRAINT `CallNum` FOREIGN KEY (`CallNum`) REFERENCES `work` (`CallNum`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `locate` (

`item` int NOT NULL,

`branch` varchar(45) DEFAULT NULL,

`rack` int DEFAULT NULL,

PRIMARY KEY (`item`),

KEY `rack\_idx` (`rack`),

KEY `branch\_idx` (`branch`),

CONSTRAINT `branch` FOREIGN KEY (`branch`) REFERENCES `branch` (`name`),

CONSTRAINT `item` FOREIGN KEY (`item`) REFERENCES `libraryitem` (`BarCode`),

CONSTRAINT `rack` FOREIGN KEY (`rack`) REFERENCES `rack` (`RackID`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `patron` (

`member\_id` int NOT NULL,

`name` varchar(45) DEFAULT NULL,

`address` varchar(45) DEFAULT NULL,

`email` varchar(45) DEFAULT NULL,

`phone\_num` int DEFAULT NULL,

PRIMARY KEY (`member\_id`),

UNIQUE KEY `card\_num\_UNIQUE` (`member\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `rack` (

`RackID` int NOT NULL,

`Section` varchar(45) NOT NULL,

`Row` int NOT NULL,

PRIMARY KEY (`RackID`),

UNIQUE KEY `UNIQUEROW` (`Section`,`Row`) COMMENT 'Combo of section/row must be unique'

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `reserve` (

`idReserve` int NOT NULL,

`Card` int NOT NULL,

`Work` varchar(20) NOT NULL,

`CreationDate` date NOT NULL,

`pickup` varchar(45) NOT NULL,

`done` tinyint NOT NULL DEFAULT '0',

PRIMARY KEY (`idReserve`),

KEY `work\_idx` (`Work`),

KEY `patron\_idx` (`Card`),

KEY `Branch\_idx` (`pickup`),

CONSTRAINT `holding` FOREIGN KEY (`Work`) REFERENCES `work` (`CallNum`),

CONSTRAINT `patron` FOREIGN KEY (`Card`) REFERENCES `patron` (`member\_id`),

CONSTRAINT `pickup\_branch` FOREIGN KEY (`pickup`) REFERENCES `branch` (`name`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `work` (

`CallNum` varchar(20) NOT NULL,

`MARC\_ID` decimal(10,0) NOT NULL,

`Format` varchar(45) DEFAULT NULL,

PRIMARY KEY (`CallNum`),

UNIQUE KEY `CallNum\_UNIQUE` (`CallNum`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci COMMENT='Library Holdings, stand in for catalogue entry';