SCS Resource Management

Requirements

Services

Catalogue Service

- This service will allow the user to search for any Resource held by the School of Computer Science using a number of different search criteria.

Loan Service

- Allows a member to Loan Resources for a period of time, according to their Privilege.

Acquisition Service

- Allows Members to request that the school acquire new Resources for their collection

Reservation Service

- Allows a Member to reserve a Resource for a particular time and date prior to when that Resource is actually required

Data Requirements

Resource

Resources are available for Student and Staff Members to use in their projects at the School of Computer Science. There are two different kinds of Resource: Movable and Immovable. Movable Resources include but are not limited to cameras, microphones, and phones. These Resources can be taken off site from the school to be used in various projects. Immovable Resources include but are not limited to classrooms and lab rooms. All Resources have a unique Resource ID, a Description of the Resource, and its Present Status. Movable Resources also keep track of Name, Make, Manufacturer, Model, Year, and Asset Value. Immovable Resources have information regarding Capacity and available Facilities. Resources are grouped in terms of their Category, and most Resources have a Location.

Category

Resources in the database are grouped in terms of their Category. Categories include but are not limited to all cameras, all microphones, and all classrooms. Categories have a unique Category ID, a Name, Description of the Resource and a Maximum Time that that type of Resource can be both Loaned or Reserved for. Members will be granted the ability to Loan or Reserve Resources of a certain Category according to their Privilege.

Location

Resources that are available at the School of Computer Science may be assigned a Location. All Immovable Resources will have a Location. Most Movable Resources will have a location, however some Resources such as digital Resources will not have a Location. Locations have a unique Location ID, a Room Number, Building, and Campus.

Member

Students and Staff at the School of Computer Science are classified as Members. Members can Loan, Reserve, and Request Resources according to their Privilege. Members are split into Staff and Student. All Members have a unique ID, Name, Address, Phone Number, Email, Status, and a Comments field. Students will also have a Demerit Points field, which will start with a value of 12 and will be decreased by 3 for every day overdue a returned Resource is, and by 1 for a Reservation that is a no-show. If the value reaches 0, the Status of the Student Member will be set to Disabled. The Administrator has the ability to manually adjust this value. Students will also be assigned Course Offerings, which dictate what Loan and Reservation Privileges are available to the Student Member. Staff members have a Role field, determining their Role at the School of Computer Science.

Course Offering

Student Members will be assigned a number of Course Offerings depending on what Courses they are taking at the School of Computer Science. A Course Offering has a level of Privileges which allow Student Members to loan Resources. A Course Offering has a unique Offering ID, the Course ID, Course Name, Semester Offered, Year Offered, Date the course begins, and Date the course concludes. If the current date is later than the Date the course concludes for all of a Student Member's Course Offerings, then their Status will be changed to Disabled.

Privilege

Students at the School of Computer Science will have a Privilege according to their Course Offerings. This Privilege dictates how many Resources of a particular Category can be borrowed at any one time. All Privileges will have a unique Privilege ID, Name, Description, Category to which the Privilege is granted for, and the maximum number of Resources that can be Loaned or Reserved at any time for that particular Category. The maximum number of allowed Resources is cumulative with multiple Privileges allowing the same Category.

Acquisition Request

If a Student or Staff member wishes to have access to a Resource that is not currently offered by the School of Computer Science, they may submit an Acquisition Request. This Request will have a unique Acquisition ID, Member requesting Acquisition, Item name, Make, Manufacturer, Model, Year, Description of the required Resource, and Urgency of the Acquisition. The urgency is a number from 1 to 10 with a 10 indicating very urgent and a 1 indicating not very urgent. The Acquisition Request may also contain additional information after it has been created to further progress the Acquisition. This information may include a Status, Fund Code, Vendor Code, Price and add any other Notes for the Acquisition. This can then be submitted to fulfill the Acquisition.

Reservation

Members have the ability to make a Reservation of a certain Resource for use at a later time. In the case of a Movable Resource, this Reservation will indicate a period of time that the Resource will be borrowed for in the form of a Loan. For an Immovable Resource, this Reservation will indicate a period of time that the Resource will be in use by the Member, and will be unavailable for other Members. Reservations can only exist for Resources that the Member has the correct Privileges to Reserve. No two reservations can conflict. A Reservation will consist of a unique Reservation ID, the Resource that will be required, the Date and Time the item is required, and the Due Date and Time (In the case of Immovable Resources, this will indicate the end time of the Reservation). Once a Reservation has been made, the Resource will be booked for the Member to Loan or use at the requested Date and Time. If a Student Member Status field is disabled, they are unable to Reserve Resources. Reservations run on a first come, first serve basis.

Loan

A Member at the School of Computer Science can borrow Resources by taking out a Loan. Loans are only available for Resources the Member has the correct Privilege to Loan. A Loan consists of the following information: Loan ID, Member lending the Resource, Date and Time loaned, Due Date and Time, Date and Time returned. If a Student Member Status field is disabled, they are unable to Loan Resources.

Transaction requirements

Data Entry

- Enter the details of a new Resource
- Enter the details of a new Category of Resources
- Enter the details of a new Member
- Enter the details of a new Location that a Resource could reside
- Initialise and enter the details of a Loan between a Member and a Resource
- Initialise and enter the details of an Acquisition Request by a Member
- Enter the details of a Reservation between a Member and a Resource
- Enter the details of a Course Offering by the School of Computer Science
- Enter the details of a Privilege that is allowed by a Course Offering that allows for certain Categories of Resources to be Loaned and Reserved

Data update/deletion

- Update/Delete the details of a Resource
- Update/Delete the details of a Category
- Update/Delete the details of a Member
- Update/Delete the details of a Location
- Update/Delete the details of a Loan
- Update/Delete the details of an Acquisition Request
- Update/Delete the details of a Reservation
- Update/Delete the details of a Course Offering
- Update/Delete the details of a Privilege

Data Queries

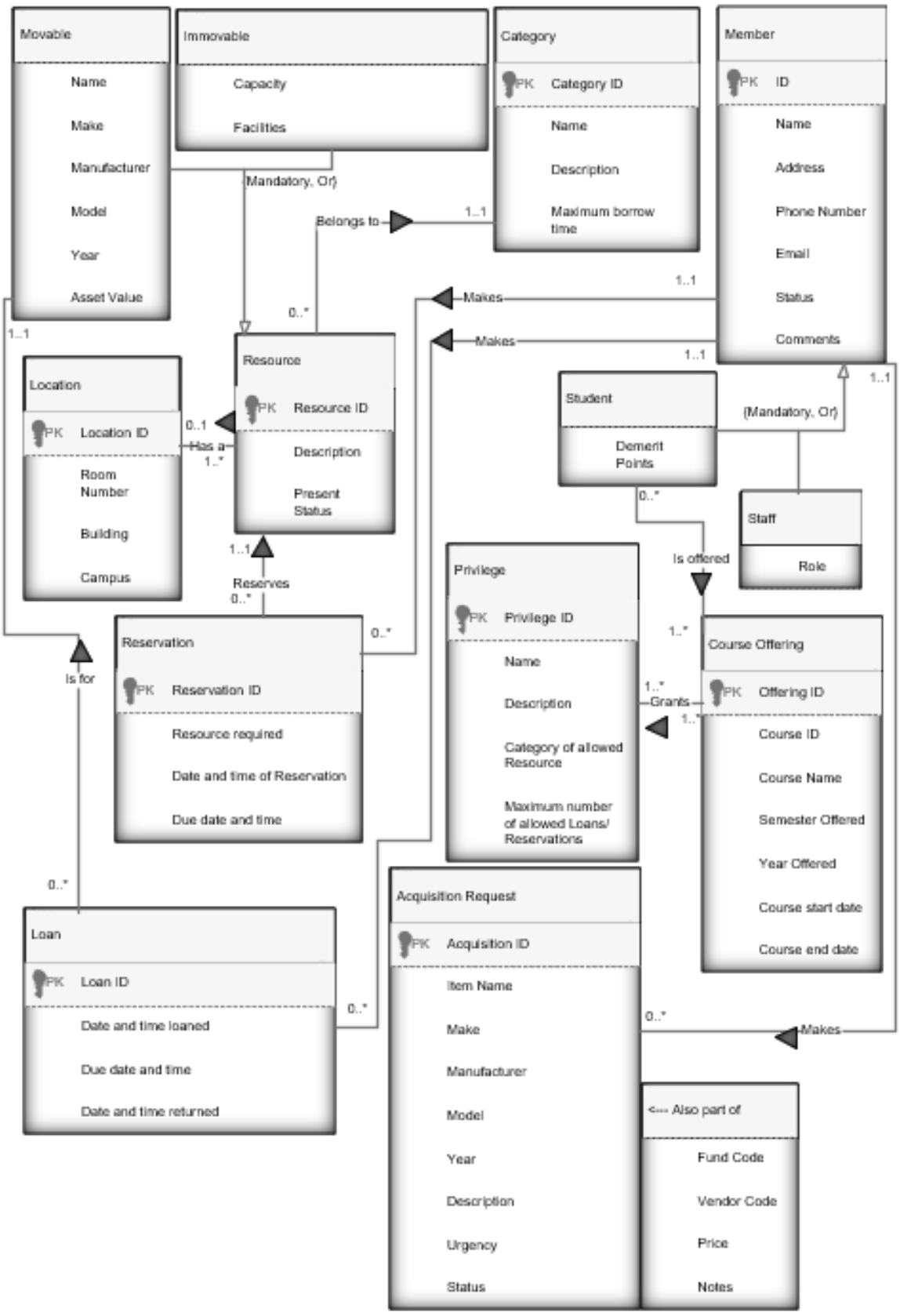
- Search a loaned item based on an ID
- Search whether a resource is owned by the SCS
- Get how many points have been earned by a student
- Get what Resources are currently available
- Get what Resources are reserved for a certain time
- Get all Acquisition Requests grouped by Urgency
- Get the Student ID of a Member who has an overdue Resource
- Get what loans are currently overdue
- Get the privileges of a student based off ID
- Get how many Resources exist for a certain Category

Business Rules

- Expiration of Student Member access
    - A Student Member's Loan and Reservation Privileges are automatically taken away when the current date is later than the end date of all of their enrolled Course Offerings
    - A Student Member's Loan and Reservation Privileges are automatically taken away when their demerit point field is reduced to 0 (see below)
    - The status of the student member is set to "Disabled"
- Maximum items Loaned or Reserved at any one time
    - A Member cannot Loan or Reserve more than the maximum number of items specified in their Privileges at any given time
    - The number of Loans or Reservations allowed by Privileges is cumulative
- Penalty for late returns by Student Members
    - Each Student Member has a default set of demerit points earned (12 to start with)
    - A penalty of 3 points is incurred for each overdue day
    - When the demerit point total is reduced to 0, member status is set to "Disabled"
    - The SCS Administrator has rights to reset/amend points
- Cancellation of Reservations
    - A Reservation of a Resource is cancelled if it is not picked up after 24 hours of the required date or due date (whichever is earlier)
    - Non cancellation of a Reservation by a Member results in a penalty of 1 point

    - When the demerit point total is reduced to 0, member status is set to “Disabled”
    - The SCS Administrator has rights to cancel any Reservation
- Loaning/Reservation Periods
    - The duration of Loaning/Reservation periods (Either number of days or hours) are determined by the category to which the item belongs, for example
        - Cameras can have a duration of 2 days
        - Microphones can have a duration of 2 hours
- Reservations cannot conflict
    - No two Reservations can be made at the same time for the same Resource
    - If there are two possible Reservations on the same Resource, the Reservation that was made first takes precedence
- Making Acquisition Requests
    - Members can submit an Acquisition Request to the SCS for Resources that are not currently owned
    - After an Acquisition Request has been made, the SCS Administrator may add further information relating to the Acquisition

Enhanced Entity-Relationship Model



Data Dictionary

Entity Types

Entity Name	Description	Aliases	Occurrence
Resource	A superclass of two different types of Resource offered by the SCS that can be Loaned or Reserved by Members	N/A	When the SCS acquires a new resource according to an acquisition request
Movable Resource	A subclass of Resource that can be moved, eg microphone, speaker etc	N/A	See Resource
Immovable Resource	A subclass of Resource that cannot be moved such as a classroom or a computer lab	N/A	See Resource
Category	A grouping of Resources of a similar type such as “All microphones” or “All studios”	N/A	When a new type of Resource is acquired by the SCS
Location	The standard location of the Resource if it is not currently Reserved or Loaned.	N/A	When a new Location is added by the SCS Administrator
Member	A superclass of two different types of people at the SCS. Holds most of the basic information about individuals	N/A	When a new member attends the SCS
Student Member	A subclass of Member with additional information pertaining to being a student at the SCS	N/A	See Member
Staff Member	A subclass of Member with additional information pertaining to being a staff member at the SCS	N/A	See Member

Course Offering	A course that is offered to Students Members that attend the SCS. Course Offerings provide Loan and Reservation Privileges to Student Members	“Course”	When the SCS administrator creates a new Course Offering
Privilege	Granted to Student Members according to Course Offerings. Affects what Categories of Resource Student Members can Reserve and Loan	N/A	When a new Course Offering is created
Acquisition Request	A request made by a Member for the SCS to acquire new Resources	N/A	When a Member makes an Acquisition Request
Reservation	A hold placed on a Resource, in the case of an Immovable Resource this being a time for the Resource to be borrowed, and for a Movable Resource this is a time for a Member to come and pick it up.	“Hold” “Book”	When a Member makes a Reservation
Loan	A Movable Resource can be Loaned to Members allowing them to take it from its Location for a given period of time according to their Privilege	“Borrow”	When a Member makes a Loan

Relationship Types

Entity Name	Multiplicity	Relationship	Multiplicity	Entity Name
Resource	Mandatory, Or	Superclass to	Mandatory, Or	Movable Resource
	Mandatory, Or	Superclass to	Mandatory, Or	Immovable Resource
	0..*	Belongs to	1..1	Category
	1..1	Is reserved by	0..*	Reservation
	1..*	Has a	0..1	Location
Movable Resource	Mandatory, Or	Subclass to	Mandatory, Or	Resource
	1..1	Can be borrowed as a	0..*	Loan
Immovable Resource	Mandatory, Or	Subclass to	Mandatory, Or	Resource
Category	1..1	Categorises/Groups	0..*	Resource
Location	0..1	Is maintained for	1..*	Resource
Member	Mandatory, Or	Superclass to	Mandatory, Or	Student
	Mandatory, Or	Superclass to	Mandatory, Or	Staff
	1..1	Makes a	0..*	Reservation
	1..1	Makes a	0..*	Loan
	1..1	Makes an	0..*	Acquisition Request
Student Member	Mandatory, Or	Subclass to	Mandatory, Or	Member
	0..*	Is offered	1..*	Course Offering
	0..*	Is granted	1..*	Privilege
Staff Member	Mandatory, Or	Subclass to	Mandatory, Or	Member
Course Offering	1..*	Offered to	0..*	Student Member
	1..*	Grants	1..*	Privilege
Privilege	1..*	Is granted to	0..*	Student Member
	1..*	Is Granted by	1..*	Course Offering
Acquisition Request	0..*	Is made by	1..1	Member
Reservation	0..*	Is made by	1..1	Member
	0..*	Reserves	1.1	Resource
Loan	0..*	Is made by	1..1	Member
	0..*	Is for	1..1	Movable Resource

Attributes

Entity Name	Attributes	Description	Data Type & Length	Nulls	Multivalued	Derived	Default
Resource	Resource ID	Unique Identifier	char(10)	N	N	N	N/A
	Description	Description of Resource	varchar(400)	Y	N	N	NULL
	Present status	Describes the status of the Resource (In use, Maintenance, Available, Loaned, Lost, Damaged, etc)	varchar(12)	Y	N	N	NULL
Movable Resource	Name	Name of Resource	varchar(30)	Y	N	N	NULL
	Make	Make of Resource	varchar(30)	Y	N	N	NULL
	Manufacturer	Manufacturer of Resource	varchar(30)	Y	N	N	NULL
	Model	Model of Resource	varchar(30)	Y	N	N	NULL
	Year	Year Resource was manufactured	Int	Y	N	N	NULL
	Asset value	Value of Resource	smallmoney	Y	N	N	NULL
Immovable Resource	Capacity	The number of Members that can be supported by the Resource	tinyint	Y	N	N	NULL
	Facilities	Facilities available from the Resource (Computers, whiteboards etc)	varchar(400)	Y	N	N	NULL
Category	Category ID	Unique Identifier	char(10)	N	N	N	N/A
	Name	Name of Category	varchar(30)	Y	N	N	NULL
	Description	Description of Category	varchar(400)	Y	N	N	NULL
	Max borrow time	Maximum borrow time for a particular type of Resource, in hours	int	Y	N	N	NULL
Location	Location ID	Unique Identifier	char(10)	N	N	N	N/A
	Room number	Room that the Resource is in	varchar(30)	Y	N	N	NULL

	Building	Building that the Resource is in	varchar(30)	Y	N	N	NULL
	Campus	Campus that the Resource is at	varchar(30)	Y	N	N	NULL
Member	ID	Unique Identifier	char(10)	N	N	N	N/A
	Name	Name of the Member	varchar(30)	Y	N	N	NULL
	Address	Address of the Member	varchar(100)	Y	N	N	NULL
	Phone number	Phone number of the Member	varchar(12)	Y	N	N	NULL
	Email	Email address of the Member	varchar(50)	Y	N	N	NULL
	Status	Either active or disabled depending on whether the member is allowed to make loans and reservations. Boolean value 0 is disabled, 1 is active	bit	Y	N	N	1
	Comments	Comments field	varchar(400)	Y	N	N	NULL
Student Member	Demerit points	These will be deducted for various rules broken, if the field reaches a value of 0, the status of the Member will be changed to disabled	tinyint	Y	N	N	NULL
Staff Member	Role	Role that the Staff Member holds at the SCS	varchar(30)	Y	N	N	NULL
Course Offering	Offering ID	Unique Identifier	char(10)	N	N	N	N/A
	Course ID	ID of the course being offered	varchar(10)	Y	N	N	NULL
	Course name	Name of the course being offered	varchar(50)	Y	N	N	NULL
	Semester offered	Semester that the course is offered (1 or 2)	tinyint	Y	N	N	NULL
	Year offered	Year that the course is offered	int	Y	N	N	NULL
	Course start date	Date that the course commences	date	Y	N	N	NULL

	Course end date	Date that the course ends. If all a student members courses have ended, their status will be changed to disabled	date	Y	N	N	NULL
Privilege	Privilege ID	Unique Identifier	char(10)	N	N	N	N/A
	Name	Name of the Privilege	varchar(30)	Y	N	N	NULL
	Description	Description of the Privilege	varchar(400)	Y	N	N	NULL
	Category of allowed resource	Category that can be Loaned and Reserved with that specific Privilege	varchar(30)	Y	N	N	NULL
	Maximum number of allowed loans/reservations	The maximum number of items allowed to be Loaned from the specified Category. Is cumulative with other Privileges allowing the same Category	int	Y	N	N	NULL
Acquisition Request	Acquisition ID	Unique Identifier	char(10)	N	N	N	N/A
	Item Name	Name of requested item	varchar(30)	Y	N	N	NULL
	Make	Make of requested item	varchar(30)	Y	N	N	NULL
	Manufacturer	Manufacturer of requested item	varchar(30)	Y	N	N	NULL
	Model	Model of requested item	varchar(30)	Y	N	N	NULL
	Year	Year requested item was manufactured	int	Y	N	N	NULL
	Description	Description of requested item	varchar(400)	Y	N	N	NULL
	Urgency	A number from 1-10 indicating the urgency of the acquisition, with 10 being urgent and 1 being not urgent	tinyint	Y	N	N	NULL
	Status	The status of the Acquisition, updated by the SCS	varchar(12)	Y	N	N	NULL

		Administrator (acquired, pending, etc)					
	Fund code	Fund code of the Acquisition	varchar(30)	Y	N	N	NULL
	Vendor code	Vendor code of the Acquisition	varchar(30)	Y	N	N	NULL
	Price	Quoted price of the Acquisition	smallmoney	Y	N	N	NULL
	Notes	Notes field	varchar(400)	Y	N	N	NULL
Reservation	Reservation ID	Unique Identifier	char(10)	N	N	N	N/A
	Resource required	Name of Resource that will be Reserved	varchar(30)	Y	N	N	NULL
	Date and time of reservation	The date and time for the Reservation to take place. For an Immovable Resource this will be the time the Resource is unavailable to other Members. For a Movable Resource this will be the pickup time for a Member to arrive and make a Loan for the Resource	datetime	Y	N	N	NULL
	Due date and time	Time that the Resource is due to be available to other Members once more	datetime	Y	N	N	NULL
Loan	Loan ID	Unique Identifier	char(10)	N	N	N	N/A
	Date and time loaned	The date and time that the Loan began	datetime	Y	N	N	NULL
	Due date and time	The date and time that the Resource is due to be returned	datetime	Y	N	N	NULL
	Date and time returned	The date and time that the Resource was returned	datetime	Y	N	N	NULL

Review on Assignment 1

Assignment 1 had little feedback to take on board, there was only minor clarifications required and the removal of a relationship between Student and Privilege (Seeing as this was implied by transitivity already), and the removal of some foreign keys from the EER such as “Member Lending Resource” in the Loan entity. After this was complete, the only thing left was to make the EER more readable which I did by changing the font, which made it a bit clearer. There were no comments to take on board in the Requirements or Data Dictionary These comments have been revised in the above document, and this revised version is what will be used for the remainder of this report.

Relational Model

Directly transferred from EER

movableResource (movableResourceID, description, presentStatus, name, make, manufacturer, model, year, assetValue, categoryID, locationID)

Primary Key movableResourceID

Foreign Key categoryID **references** category(categoryID) on **update** cascade on **delete** do nothing

Foreign Key locationID **references** location(locationID) on **update** cascade on **delete** do nothing

immovableResource (immovableResourceID, description, presentStatus, capacity, facilities, categoryID, locationID)

Primary Key immovableResourceID

Foreign Key categoryID **references** category(categoryID) on **update** cascade on **delete** do nothing

Foreign Key locationID **references** location(locationID) on **update** cascade on **delete** do nothing

category (categoryID, name, description, maxBorrowTime)

Primary Key categoryID

location (locationID, roomNumber, building, campus)

Primary Key locationID

studentMember (studentID, name, address, phoneNumber, email, status, comments, demeritPoints, studentOfferingID)

Primary Key studentID

staffMember (staffID, name, address, phoneNumber, email, status, comments, role)

Primary Key staffID

courseOffering (courseOfferingID, courseID, courseName, semesterOffered, yearOffered, courseStartDate, courseEndDate)

Primary Key offeringID

Alternate Key courseID

privilege (privilegeID, name, description, resourceCategory, maxLoans)

Primary Key privilegeID

studentReceivesCourseOffering(studentOfferingID, courseOfferingID)

Primary Key studentOfferingID, courseOfferingID

Foreign Key studentOfferingID **references** studentMember(studentID) on **update** cascade on **delete** do nothing

Foreign Key courseOfferingID **references** courseOffering (offeringID) on **update** cascade on **delete** do nothing

courseOfferingGrantsPrivilege (courseOfferingID, privilegeID)

Primary Key courseOfferingID, privilegeID

Foreign Key courseOfferingID **references** courseOffering(courseOfferingID) on **update** cascade on **delete** cascade

Foreign Key privilegeID **references** privilege(privilegeID) on **update** cascade on **delete** cascade

acquisitionRequest (acquisitionID, itemName, make, manufacturer, model, year, description, urgency, status, fundCode, vendorCode, price, notes, studentID, staffID)

Primary Key acquisitionID

Foreign Key studentID **references** studentMember(studentID) on **update** cascade on **delete** cascade

Foreign Key staffID **references** staffMember(staffID) on **update** cascade on **delete** cascade

reservation (reservationID, reserveDateTime, dueDateTime, studentID, staffID, movableResourceID, immovableResourceID)

Primary Key reservationID

Foreign Key studentID **references** studentMember(studentID) on **update** cascade on **delete** cascade

Foreign Key staffID **references** staffMember(staffID) on **update** cascade on **delete** cascade

Foreign Key movableResourceID **references** movableResource(movableResourceID) on **update** cascade on **delete** cascade

Foreign Key immovableResourceID **references** immovableResource(immovableResourceID) on **update** cascade on **delete** cascade

loan (loanID, loanedDateTime, dueDateTime, returnedDateTime, studentID, staffID, movableResourceID)

Primary Key loanID

Foreign Key studentID **references** studentMember(studentID) on **update** cascade on **delete** cascade

Foreign Key staffID **references** staffMember(staffID) on **update** cascade on **delete** cascade

Foreign Key movableResourceID **references** movableResource(movableResourceID) on **update** cascade on **delete** cascade

Normalisation Process

By definition of the relational model, all of these relations are in 1NF

They are made up of single atomic values, columns have unique names, and the data in each column is of the same domain

They are also in 2NF, as there are no partial dependencies

(A partial dependency is where there are two prime attributes, and an attribute in the table is only dependent on one of these attributes but not both. In this case the table should be split into two tables)

Most are also in 3NF as there are no transitive dependencies, and where they are not, they are specified and normalized below

(A transitive dependency is where X, Y and Z are in a table and X implies Y, and Y implies Z. As such it can be inferred that X implies Z, and the table should be split to prevent redundant data)

Most are also in BCNF as the only functional dependency is a Primary Key. Where they are not they are normalised below

movableResource

Table is in 2nd Normal Form according to the definition above

FD1: movableResourceID → presentStatus, categoryID, modelID

FD2: modelID → model, make, manufacturer, description, name, year, assetValue

Transitive Dependency, movableResourceID → modelID and modelID → resource details. Split into two tables to remove transitive dependency. This assumes that the model number is unique among all items owned by the SCS.

Table 1:

movableResource (movableResourceID, presentStatus, categoryID, modelID, locationID)

Primary Key movableResourceID

Foreign Key categoryID **references** category(categoryID) on **update** cascade on **delete** do nothing

Foreign Key modelID **references** resourceDetails(modelID) on **update** cascade on **delete** cascade

Foreign Key locationID **references** location(locationID) on **update** cascade on **delete** do nothing

movableResource is now in 3NF as it has no transitive dependencies. The relation is also in BCNF as the primary key is the only functional dependency.

Table 2:

resourceDetails(modelID, model, make, manufacturer, description, name, year, assetValue)

Primary Key modelID

resourceDetails is now in 3NF as it has no transitive dependencies. The relation is also in BCNF as the primary key is the only functional dependency.

immovableResource

The relation is in BCNF. The only functional dependency is the Primary Key. This assumes that there is at least one instance in which a room has exactly the same capacity and facilities as another.

Category

The relation is in BCNF. The only functional dependency is the Primary Key. The max borrow time for any Resource could match the max borrow time for any other Resource quite easily

Location

The relation is in BCNF. The only functional dependency is the Primary Key. This assumes that room numbers are generic, and there can be rooms with the same number that are in different buildings.

studentMember

The relation is in BCNF. The only functional dependency is the Primary Key. This assumes that there may be instances where Student Members may share an email or phone number with another Student Member such as in the case of siblings.

staffMember

The relation is in BCNF. The only functional depend is the Primary Key. This assumes that there may be instances where Staff Members may share an email or phone number with another Staff Member such as in the case of siblings or spouses.

courseOffering

Table is in 2nd Normal Form according to the definition above

FD1: offeringID → courseID, courseName, semesterOffered, yearOffered

FD2: semesterOffered → courseStartDate, courseEndDate

Transitive Dependency, offeringID → courseID and courseID → semester details. Split into two tables to remove transitive dependency.

Table 1:

courseOffering (offeringID, courseID, courseName, yearOffered, semesterOffered)

Primary Key offeringID

Alternate Key courseID

Foreign Key semesterOffered **references** semesterDetails(semesterOffered) on **update** cascade on **delete** do nothing

courseOffering is now in 3NF as it has no transitive dependencies. The relation is also in BCNF as the primary key is the only functional dependency.

Table 2:

semesterDetails (semesterOffered, courseStartDate, courseEndDate)

Primary Key semesterOffered

The relation is now in 3NF and also in BCNF. The only functional dependencies are Primary Keys.

semesterDetails is now in 3NF as it has no transitive dependencies. The relation is also in BCNF as the primary key is the only functional dependency.

Privilege

The relation is in BCNF. The only functional depend is the Primary Key. This assumes that there is more than one privilege that grants access to the same category, although it may allow for a different maximum number of Resources Borrowed.

studentReceivesCourseOffering

The relation is in BCNF. The only attributes in the relation are the PK as it is a reference table.

courseOfferingGrantsPrivilege

The relation is in BCNF. The only attributes in the relation are the PK as it is a reference table.

acquisitionRequest

Relation is in 2nd Normal Form according to the definition above

FD1: acquisitionID → urgency, status, fundCode, vendorCode, notes, studentID, staffID, modelID

FD2: modelID → model, make, manufacturer, description, name, year, price

FD2 for acquisitionRequest is the same as FD2 for movableResource, with the only difference being "price" vs "assetValue". As such, the "price" column in acquisitionRequest will be renamed "assetValue" and the same table can be used. This will make for more efficient database usage when the SCS acquires the asset, as a lot of the information is already stored in the database.

Transitive Dependency, acquisitionID → modelID and modelID → acquisition details. This assumes that the student or staff member are able to make more than one acquisition at a time, and that the model number, is unique among all items owned by the SCS. Split into two tables to remove transitive dependency.

Table 1:

acquisitionRequest (acquisitionID, urgency, status, fundCode, vendorCode, notes, studentID, staffID, modelID)

Primary Key acquisitionID

Foreign Key studentID **references** studentMember(studentID) on **update** cascade on **delete** cascade

Foreign Key staffID **references** staffMember(staffID) on **update** cascade on **delete** cascade

Foreign Key modelID **references** resourceDetails(modelID) on **update** cascade on **delete** cascade

acquisitionRequest is now in 3NF as it has no transitive dependencies. The relation is also in BCNF as the primary key is the only functional dependency.

Table 2:

resourceDetails(modelID, model, make, manufacturer, description, name, year, assetValue)

Primary Key modelID

As described in movableResource, resourceDetails is in 3NF as it has no transitive dependencies. The relation is also in BCNF as the primary key is the only functional dependency.

Reservation

Relation is in 2nd Normal Form according to the definition above

FD1: reservationID → reserveDateTime, resourceRequiredID, studentID, staffID

FD2: resourceRequiredID → dueDateTime, movableResourceID, immovableResourceID

Transitive Dependency, reservationID → resourceRequiredID and resourceRequiredID → resource required details. Split into two tables to remove transitive dependency.

Table 1:

reservation (reservationID, reserveDateTime, studentID, staffID, resourceRequiredID)

Primary Key reservationID

Foreign Key studentID **references** studentMember(studentID) on **update** cascade on **delete** cascade

Foreign Key staffID **references** staffMember(staffID) on **update** cascade on **delete** cascade

Foreign Key resourceRequiredID **references** resourceRequired (resourceRequiredID) on **update** cascade on **delete** cascade

reservation is now in 3NF as it has no transitive dependencies. The relation is also in BCNF as the primary key is the only functional dependency.

Table 2:

resourceRequired (resourceRequiredID, dueDateTime, movableResourceID, immovableResourceID)

Primary Key resourceRequiredID

Foreign Key movableResourceID **references** movableResource(movableResourceID) on **update** cascade on **delete** cascade

Foreign Key immovableResourceID **references** immovableResource(immovableResourceID) on **update** cascade on **delete** cascades

resourceRequired is now in 3NF as it has no transitive dependencies. The relation is also in BCNF as the primary key is the only functional dependency.

loan

Table is in 2nd Normal Form according to the definition above

FD1: loanID → loanedDateTime, returnedDateTime, studentID, staffID, resourceRequiredID

FD2: resourceRequiredID → dueDateTime, movableResourceID

FD2 for loan is the same as FD2 for reservation, with the only difference being the extra column "immovableResourceID" in reservation. As such, the same table can be used. This will make for more efficient database usage when a reservation becomes a loan.

Transitive Dependency, loanID → resourceRequiredID and resourceRequiredID → resource required details. Split into two tables to remove transitive dependency.

Table 1:

loan (loanID, loanedDateTime, returnedDateTime, studentID, staffID, resourceRequiredID)

Primary Key loanID

Foreign Key studentID **references** studentMember(studentID) on **update** cascade on **delete** cascade

Foreign Key staffID **references** staffMember(staffID) on **update** cascade on **delete** cascade

Foreign Key movableResourceID **references** movableResource(movableResourceID) on **update** cascade on **delete** cascade

Foreign Key resourceRequiredID **references** resourceRequired on **update** cascade on **delete** cascade

loan is now in 3NF as it has no transitive dependencies. The relation is also in BCNF as the primary key is the only functional dependency.

Table 2:

resourceRequired (resourceRequiredID, dueDateTime, movableResourceID, immovableResourceID)

Primary Key resourceRequiredID

Foreign Key movableResourceID **references** movableResource(movableResourceID) on **update** cascade on **delete** cascade

Foreign Key immovableResourceID **references** immovableResource(immovableResourceID) on **update** cascade on **delete** cascades

As described in reservation, resourceRequired is in 3NF as it has no transitive dependencies. The relation is also in BCNF as the primary key is the only functional dependency.

Normalised Relational Schema

movableResource (movableResourceID, presentStatus, categoryID, modelID, locationID)

Primary Key movableResourceID

Foreign Key categoryID **references** category(categoryID) on **update** cascade on **delete** do nothing

Foreign Key modelID **references** resourceDetails(modelID) on **update** cascade on **delete** cascade

Foreign Key locationID **references** location(locationID) on **update** cascade on **delete** do nothing

resourceDetails(modelID, model, make, manufacturer, description, name, year, assetValue)

Primary Key modelID

immovableResource (immovableResourceID, description, presentStatus, capacity, facilities, categoryID, locationID)

Primary Key immovableResourceID

Foreign Key categoryID **references** category(categoryID) on **update** cascade on **delete** do nothing

Foreign Key locationID **references** location(locationID) on **update** cascade on **delete** do nothing

category (categoryID, name, description, maxBorrowTime)

Primary Key categoryID

location (locationID, roomNumber, building, campus)

Primary Key locationID

studentMember (studentID, name, address, phoneNumber, email, status, comments, demeritPoints, offeringID)

Primary Key studentID

staffMember (staffID, name, address, phoneNumber, email, status, comments, role)

Primary Key staffID

courseOffering (offeringID, courseID, courseName, yearOffered, semesterOffered)

Primary Key offeringID

Alternate Key courseID

Foreign Key semesterOffered **references** semesterDetails(semesterOffered) on **update** cascade on **delete** do nothing

semesterDetails (semesterOffered, courseStartDate, courseEndDate)

Primary Key semesterOffered

privilege (privilegeID, name, description, resourceCategory, maxLoans)

Primary Key privilegeID

studentReceivesCourseOffering(studentOfferingID, offeringID)

Primary Key studentOfferingID, offeringID

Foreign Key studentOfferingID **references** studentMember(studentID) on **update** cascade on **delete** do nothing

Foreign Key offeringID **references** courseOffering (offeringID) on **update** cascade on **delete** do nothing

courseOfferingGrantsPrivilege (offeringID, privilegeID)

Primary Key offeringID, privilegeID

Foreign Key offeringID **references** courseOffering(offeringID) on **update** cascade on **delete** do nothing

Foreign Key privilegeID **references** privilege(privilegeID) on **update** cascade on **delete** do nothing

acquisitionRequest (acquisitionID, urgency, status, fundCode, vendorCode, notes, studentID, staffID, modelID)

Primary Key acquisitionID

Foreign Key studentID **references** studentMember(studentID) on **update** cascade on **delete** cascade

Foreign Key staffID **references** staffMember(staffID) on **update** cascade on **delete** cascade

Foreign Key modelID **references** resourceDetails(modelID) on **update** cascade on **delete** cascade

reservation (reservationID, reserveDateTime, studentID, staffID, resourceRequiredID)

Primary Key reservationID

Foreign Key studentID **references** studentMember(studentID) on **update** cascade on **delete** cascade

Foreign Key staffID **references** staffMember(staffID) on **update** cascade on **delete** cascade

Foreign Key resourceRequiredID **references** resourceRequired (resourceRequiredID) on **update** cascade on **delete** cascade

resourceRequired (resourceRequiredID, dueDateTime, movableResourceID, immovableResourceID)

Primary Key resourceRequiredID

Foreign Key movableResourceID **references** movableResource(movableResourceID) on **update** cascade on **delete** cascade

Foreign Key immovableResourceID **references** immovableResource(immovableResourceID) on **update** cascade on **delete** cascade

loan (loanID, loanedDateTime, returnedDateTime, studentID, staffID, resourceRequiredID)

Primary Key loanID

Foreign Key studentID **references** studentMember(studentID) on **update** cascade on **delete** cascade

Foreign Key staffID **references** staffMember(staffID) on **update** cascade on **delete** cascade

Foreign Key movableResourceID **references** movableResource(movableResourceID) on **update** cascade on **delete** cascade

Foreign Key resourceRequiredID **references** resourceRequired on **update** cascade on **delete** cascade

Review on Assignment 2

Assignment 2 feedback was mostly issues regarding the normalisation. With an extensive drafting process, the tables that were in 2NF were normalised to 3NF, which in this particular instance also meant that they were in BCNF by default. The link between courseOffering and Privilege was also adjusted, with an extra reference table added. The final tables were relatively unchanged, with the addition of the extra reference table. These have been revised in the above document, and this revised version is what will be used for the remainder of this report.