**CALIFORNIA STATE UNIVERSITY, SAN BERNARDINO**

**SCHOOL OF COMPUTER SCIENCE & ENGINEERING**

**Database Design Project**

**CSE 572 Spring 2019**

**PROJECT OPTION 2**

***University Database***

**TEAM NAME**

**Motes**

**TEAM MEMBERS**

**Vargas, Richard**

**Pulido, Justin**

**Project Description:**

You have been approached by a University for the design and implementation of a relational database system that will provide information on the courses it offers, the academic departments that run the courses, the academic staff and the enrolled students. The system will be used mainly by the students and the academic staff.

The requirement collection and analysis phase of the database design process provided the following data requirements for the University Database system.

Each department runs a number of courses. The university provides a set of modules used in different courses. Each course uses a number of modules but not every module is used. A course is assigned a unique course code and a module is identified by a unique module code. A module can be used in one course only, but can be studied by many students. In addition to the module code each module unique title, start date, end date, texts (books), and assessment scheme (i.e. coursework and exam marks percentages) are also stored.

Each course is managed by a member of academic staff, and each module is coordinated by a member of academic staff also. The database should also store each course unique title, and duration (in years).

A student can enroll in one course at a time. Once enrolled a student is assigned a unique matriculation number. To complete a course, each student must undertake and pass all the required modules in his/her course. This requires that the database store the performance (pass or fail) of each student in every module.

Additional data stored on each student includes student name (first and last), address (town, street, and post code), date-of-birth, sex, and financial loan. For emergency purposes the database stores the name (not composite), address (not composite), phone, and relationship of each student next-of-kin. None of the next-of-kin's attributes is unique. Assume that every next-of-kin is a next-of-kin of one student only.

Each department is managed by a member of academic staff. The database should record the date he/she started managing the department. Each department has a name, phone number, fax number, and location (e.g. E Block). Each department employs many members of academic staff.

A member of academic staff can be the leader (i.e. manager) of at most one course, but can be the coordinator of more than one module. A member of academic staff may not be assigned any of the above mentioned roles (coordinator, course leader, department manager). All members of academic staff teach modules. Every member of academic staff teaches one or more modules, and a module may be taught by more than one member of academic staff. The database should record the number of hours per week a member of academic staff spend teaching each module. Each member of academic staff is identified by a unique staff number. All members of staff and students have unique computer network user ID numbers. Additional data stored on each member of academic staff includes name (first and last), phone extension number, office number, sex, salary, post (lecturer, or senior lecturer, or Professor, etc.), qualifications, and address (not composite). A member of academic staff work for one department only.

**PART I: Conceptual Model**

**Mission Statement:**

The mission of the project is to provide a database that will be able to show the academic department and what courses they are offering. As well as providing information on the courses themselves along with the academic staff and the students that are enrolled.

**Data Requirements:**

1. **Academic Department**

The University has one Academic Department that will hold a majority of the information. The department is allocated members of staff, including a department manager, who manages that specific department. This includes phone numbers, fax numbers, location and the full name of the department and who works for the department with the person who manages. Additional data is held on the manager of the department regarding how long they have been the manager at that specific department. Also, the Academic department employs the staff and will have record of them as well. Also having records of the courses that the department offers.

2. **Staff**

The Staff is able to manage the Academic Department if chosen to do so. They will also have a network ID, full name, phone extension, office number, sex, salary, post, qualifications and their address on record. The staff will also be responsible for teaching the modules and keeping track of their hours. As well as managing the courses as the professor is able to teach one course at a time.

3. **Modules**

The modules will contain specific information regarding the code for the course, the title of the module, start and end date, Text books, and Assessment Scheme. This will then be used to evaluate the students in terms of a pass or fail for a grade. It will also be linked to the courses as the courses use this module. As well as previously mentioned, the modules are also linked to the staff that teach the modules.

4. **Students**

Students will have a network ID, full name, address, DOB, Sex, and Financial Loans (if any). This will all be stored on the database along with a weak entity known as Emergency Information that will hold their phone number, address, full name and NOK. The students will also be able to enroll into the courses and be given a Matriculation Number for enrollment into the course. They will then receive a pass or fail from them studying the modules in the course that is taken.

5. **Courses**

Courses will then have a course code, course title, pre requisites, and the duration of course itself. It will also contain the modules within the course such as the code number, the title of it, when does it start and end, text books that are required, and Assessment Scheme of the course itself. It will then have each of these requirements stored on the database for these requirements of it.

**Transaction Requirements:**

**1. Data Entry**

* Enter details of the department
* Enter details who is managing the department from staff.
* Enter which staff is currently teaching under the specific department.
* Enter details regarding the staff and their information when they are employed.
* Enter the courses that the department manages.
* Enter the details of that course from the modules themselves.
* Enter the details regarding who is teaching the modules and which professor is managing the courses.
* Enter student’s information once they are enrolled into the university and enter their emergency information.
* Students allowed to enroll into the courses once they are enrolled in the university.

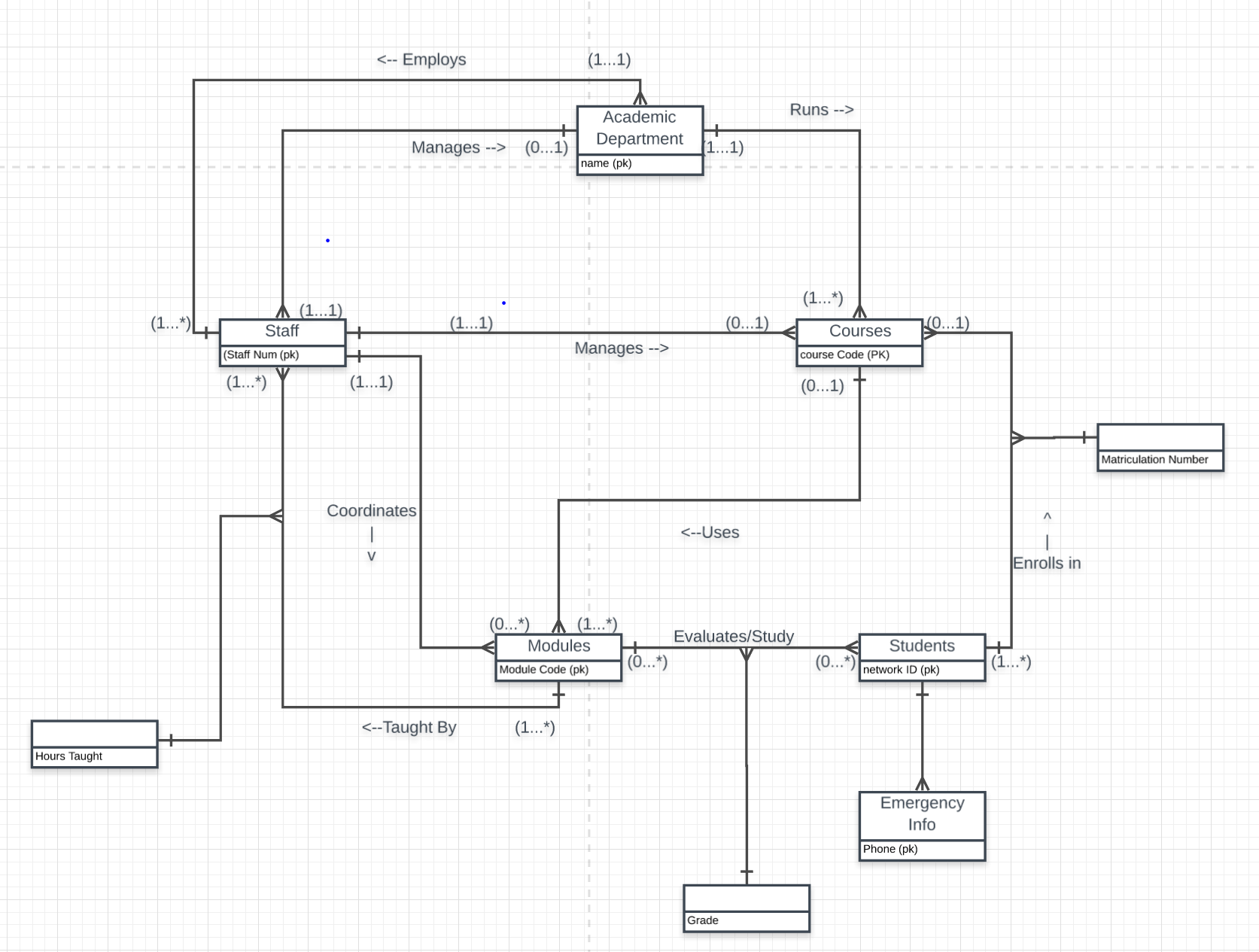
**2. Data update/deletion**

* Update/Delete the details of the Academic Department
* Update/Delete the details of a member of the staff that is managing the Department.
* Update/Delete details of the student if changed or if they leave.
* Update/Delete details of the courses once they are entered or when they are offered again.
* Update/Delete details of staff.
* Update/Delete details of modules.

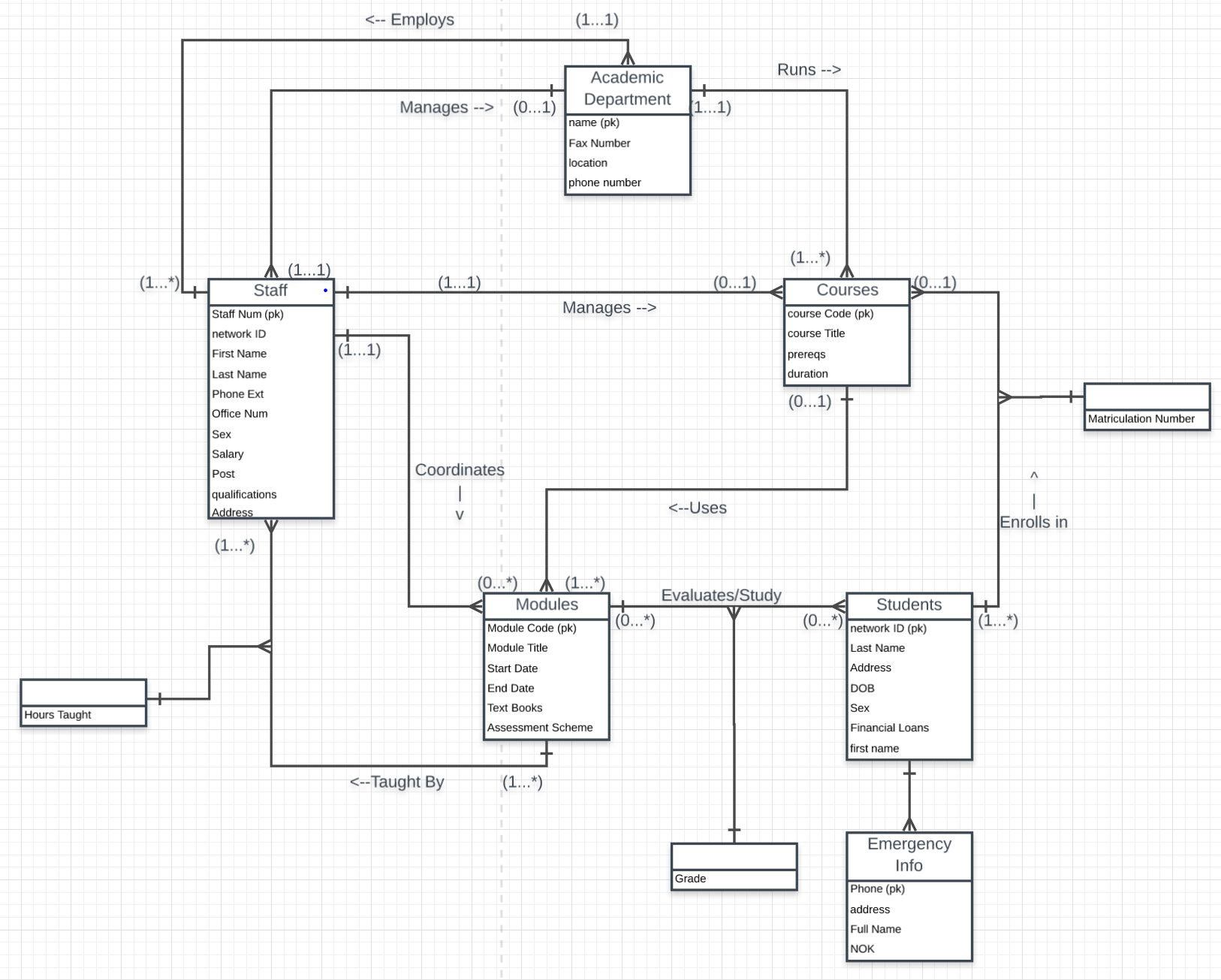
**3. Data Queries**

1. List details of the Academic Department .
2. Identify all Academic Departments in the University.
3. List name, position, and salary of staff at the Academic Department, ordered by staff name.
4. Identify the total number of staff and the sum of their salaries.
5. Identify who has worked as the department manager from the staff on records.
6. List the courses that are currently being offered by the Academic Department.
7. List all data regarding the staff.
8. Identify which staff manages the courses and which modules are coordinated by staff.
9. List the number of hours each staff has been teach for that specific module.
10. List the staff in order by name.
11. Identify if any staff is not teaching a module/course.
12. List the contents of the courses that are required from the given modules.
13. List pre requisites for the given course and how long it is in the given week.
14. Identify how many students are enrolled in the course.
15. List who is enrolled in the courses.
16. List the matriculation number given to the students who are enrolled in the course.
17. Identify how many students are currently enrolled in the university.
18. List student’s information such as Network ID, full name, DOB, etc.…
19. List student’s Emergency information if needed.
20. List details of which courses students have taken and their grades they have received.
21. List details of how many courses a student has taken.
22. List details of modules.
23. Identify the modules that are given to the courses.
24. Identify how many modules there are.
25. List if the module is in use or not in use at the given time.
26. Identify which courses are using which modules.

**G) Conceptual Model Part A – Relationship/Entities**



**Conceptual Model Part B – Entities/Attributes**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Entity Name | Attributes | Description | Data Type & Length | Null? |
| Academic Department | name {PK} | Unique Department Name | char(5) | Y |
|  | phoneNumber | Department Telephone Number | char(10) | Y |
|  | faxNumber | Department Fax Number | char(10) | N |
|  | location |  |  |  |
|  | strAddress | Street Address | varchar2(20) | Y |
|  | city | City | varchar2(15) | Y |
|  | postalCode | Postal Zip Code | char(5) | Y |
|  |  |  |  |  |
| Courses | courseCode {PK} | Unique Course Code | char(5) | Y |
|  | courseTitle | Course Title | varchar2(15) | Y |
|  | prereqs | Prerequisite Courses | varchar2(45) | N |
|  | duration | Duration in days | number(3) | N |
|  |  |  |  |  |
| Staff | staffNum {PK} | Unique Staff Number | char(5) | Y |
|  | networkID | Unique Network ID | char(5) | Y |
|  | firstName | First Name | varchar2(12) | Y |
|  | lastName | Last Name | varchar2(12) | Y |
|  | phoneExt | Phone extension | char(4) | N |
|  | officeNum | Office Number | char(10) | Y |
|  | sex | Sex | char(1) | N |
|  | salary | Salary | number(8,2) | Y |
|  | post | Position | varchar2(10) | N |
|  | qualifications | Qualifications | varchar2(30) | Y |
|  | address |  |  |  |
|  | strAddress | Street Address | varchar2(20) | Y |
|  | city | City | varchar2(15) | Y |
|  | postalCode | Postal Zip Code | char(5) | Y |
|  |  |  |  |  |
| Modules | moduleCode {PK} | Unique Module Code | char(5) | Y |
|  | moduleTitle | Module Title | varchar2(10) | Y |
|  | startDate | Start Date | date | N |
|  | endDate | End Date | date | N |
|  | textBooks | Required Text Books for Module | varchar2(30) | N |
|  | assessmentScheme | Assesment Plan | varchar2(100) | Y |
|  |  |  |  |  |
| Students | networkID {PK} | Unique Network ID | char(5) | Y |
|  | firstName | First Name | varchar2(12) | Y |
|  | lastName | Last Name | varchar2(12) | Y |
|  | DOB | DOB | date | Y |
|  | sex | Sex | char(1) | N |
|  | financialLoans | Financial Loans | number(8,2) | N |
|  |  |  |  |  |
| EmergencyInfo | Phone {PK} | Unique telephone number | char(10) | N |
|  | address | Full address | varchar2(40) | Y |
|  | fullName | First and last name | varchar2(24) | N |
|  | NOK | Next of Kin | varchar2(24) | Y |

**H) Data Dictionary Representation:**

**J) Business Rules:**

1. Academic Department:

* Each Academic Department is managed by one manager, who is an Academic Staff.
* Each Academic Department employs staff and can have many staff and can run many offer many courses.
* Each Academic Department has their own name and their own telephone line.
* Each Academic Department hires staff as well.

2. Academic Staff:

* Staff can either be manager of Academic Department or manages courses.
* Each Staff is registered with the Academic Department and can only be Registered in one Department at a time.
* Each Academic Staff has an office number as well as a set of courses they are teaching.
* Each Academic Stuff will also have their qualifications before being able to register with the Academic Department.

3. Students:

* Students are registered with the university and give their information for emergency information, as well as receiving specific information regarding their credentials.
* Students are able to enroll into a course and able to study a module and receive a grade for. (Pass or fail)
* Students are given a matriculation number once they enroll into a course and they are able to view it.
* Students can only enroll in one course at a time.
* Students are able to view all course details.

4. Courses:

* Courses are given a course code form the module which are the primary key for it.
* Courses will also have title, but some courses could have the same course title but each one will have a different code.
* It will also list the pre requisites for the course as well.
* It will also give the duration of the course.
* Courses will also be linked to Students, Modules and to Academic Department.

5. Modules:

* Modules will be linked to the courses to be used.
* Modules will have a module code, a title, start and end date, text books, and assessment scheme for the courses.
* Module are for the student to study and they will be evaluated based on their performance on this specific module.
* Module will be coordinated by staff but does not have to be.
* Each module must be taught by at least 1 staff.

**K) Common Comments – Conceptual Model Phase**

**Justin:**

1. The difficulties that I had facing this project was condensing it to be a way for it to be more understandable in creating the diagram and writing what the specific data requirements of each component. Another was missing one entity that my partner corrected me on was the modules. I had forgotten to put the module in the ER diagram and was puzzled why I need it and he explained why it needs to be there. The first difficulty was resolved by breaking the section up into smaller portions and seeing that certain portions were connected to others.
2. The likes about this project is seeing how everything is connected to specific parts of the entities. This has helped me gain a better grasp of how the database will interact with certain level of users.
3. The most challenging portion of this project for me was the Data Dictionary Representation. Still not fully understanding it but for the most part it is just creating a list of tables that will hold specific information regarding it.
4. A suggestion would be to have students create a rough draft of the document itself and have them present it to you so that they are able to see if they are on understanding the requirements fully or if they need to change specific portions of it.

**Richard:**

1. The biggest difficulty during the implementation phase was taking the written specifications and transforming them into the ER diagram. Understanding what the entities are and the relationship between them can be difficult when its written in a pure English description, without any technical notes attached, however I see the importance in this as working with those who are not technically skilled, this would be what one would encounter. Over coming this issue was helpful by collaborating with my partner, who is also technically skilled, and seeing if he agrees with my interpretation of the description and agrees with implementation.
2. Mostly everything about this portion of the project was enjoyable, however my one complaint would have to be the amount of documentation that is involved with UML. Not that UML is unimportant, but it is very tedious and time consuming to create, typically if you need to redo a small portion of the diagram, you must redo the entire diagram, since most of the software used if unforgiving in that regard.
3. The most challenging aspect, as I stated earlier, would have to be the portion of translating user needs into a technical schematic to build off of, in this case the ER diagram. It takes a few reads over, some collaboration with others, as well as a good understanding of relationships of the entities.
4. Maybe assigning specific jobs to team members, such as a “liaison” to the nontechnical users who would create the requirements, the mission statement, the transaction requirements, and then another job for the other team member, who may create UML, data dictionary, and define business rules. Just so there is clear definition of what each person must do.

**I. Personal Comments on Project – Conceptual Model Phase**

**A) Justin**

* The portions I mainly contributed too was the writing of the Project Specifications as well as identifying the PK’s of the entities in the EER diagram.
* My partner Richard contributed to creating the EER diagram and having the first diagram and in helping me understand specific portions of the Project Specifications. As well as overlooking more of the document and seeing my mistake in mixing up the two portions.

**B) Richard**

* The portions I contributed mainly to the definition of entities, relationships, creation of ER diagram, and defining the data dictionary.
* Justin contributed by writing up the main document for phase one, as well as creating EER with all attribute definition, defining primary keys, and assisting in anything I may have no understood about the project, as well as having him double check any work I had done for the project.