IV1351: Seminar 1

Pontus Åhlin

November 2023

1 Introduction

The main requirement in this assignment is to create a conceptual model for a imaginary school called The Soundgood Music School. In a free flowing text they state what are to include in the conceptual model, such as what data is to be stored for the students, instructors, the payment for the student and instructor and also that students may borrow instruments. The conceptual model is to be made to only store data that is written from the free flowing text.

For this seminar I worked alone.

The whole conceptual model is quite large as I used Astah's inbuilt the auto layout function and it makes quite a lot of space between the entities. So only specific parts will be shown in the result section. The full conceptual model is in my git repository: `https://github.com/PontusAhlin/IV1351/tree/main/Seminar%201`

2 Literature Study

I started the study with the Introduction to Databases which gave a good overview of what we as students are to accomplish during this course. Thereafter I watched all of the videos that were uploaded for module 1 on the canvas page. They gave a good understanding to how a domain model which is linked to reality is made. But also a very good insight to how one should start working on the conceptual model, what an entity, attribute and relation is and how they are connected to the conceptual model. While making the conceptual model there were multiple times where I had some doubts if I was following the right conventions, such as how to use correct IE notations and if I should make a class without attributes. In those cases I turned back to the videos. And if that wasn't enough I used the book and Google.

3 Method

First thing in the assignment was the find a way to get all of the information from the described text into a conceptual model. What I used for that was

Noun-identification. The next step after identifying all of the nouns was to make all of them separate entities into a database drawing tool. The tool used for this was Astah. For the part where we are to use the category list, I skipped it due to the statement in the requirements where it says *"The database must store all data described above, in sections 1.1 and 1.2,* ***but no other data****"*. Thus it felt unnecessary to start working with the category list, even though I understand the purpose for it. Then I started with the separation of what should be a entity and what should be an attribute of those entities. While separating everything I put all information that was related for example the student, next to the student. I did the same thing for the Instructor, the payments and the rest of the requirements. Numerous times I read through the requirement text to make sure that when an entity or attribute was removed, it was most likely redundant. When I had enough entities I started putting relations to everything. While doing that I also made comments about specific requirements that couldn't be put into the conceptual model as entities or attributes.

4 Result

The **Noun-identification** process was just me reading all of Soundgood Music School requirements and picking out the word when I saw that it was a noun. There were a couple of words that were Nouns but they weren't useful or just irrelevant. Such as: **flexibility** or **Wind/String**. I found there to be around 38 nouns that were useful in the conceptual model. I made 38 entities with these nouns. Separating the entities into attributes was a fairly straightforward process. I used a guideline that was mentioned in the videos that if we have a String, int, Boolean or time, it's most likely an attribute.

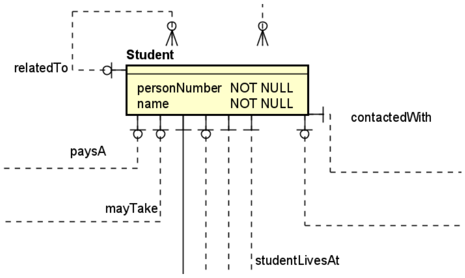


Figure 1: A student entity from the CM

Which can be seen in this case where the student should of course be an entity itself and its attributes being personNumber and name. Where personNumber is an int and name is a String. This type of thinking I used when working on the entities attributes. As can be seen above that's how I made the student entity, for the contact details I had a separate entity because it's to be used for the Instructor as well as the Enrollment entities.

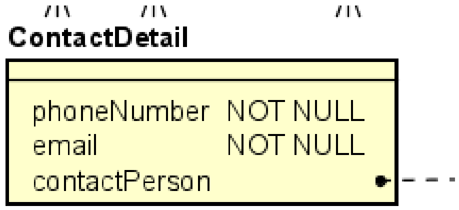


Figure 2: Contact details that are used for the Student, Instructor and Enrollment entities

The Instructor entity is made up of some of the same criteria as the student, except that it also have specified instruments that they can teach and also the proficiency check that they can teach an ensemble.

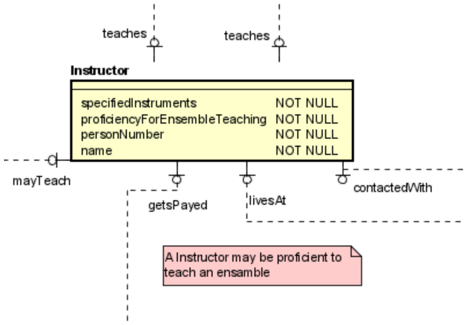
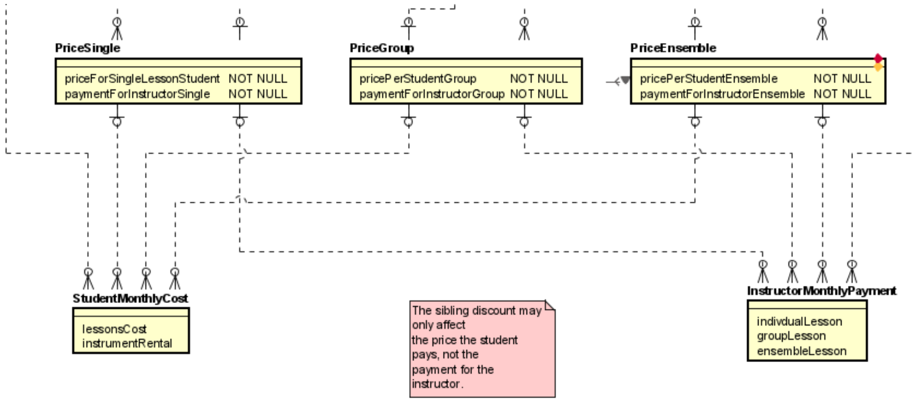

teaches
teaches
Instructor
specifiedInstruments NOT NULL
proficiencyForEnsembleTeaching NOT NULL
personNumber NOT NULL
name NOT NULL
mayTeach
getsPayed
livesAt
contactedWith
A Instructor may be proficient to teach an ensamble

Figure 3: Instructor entity

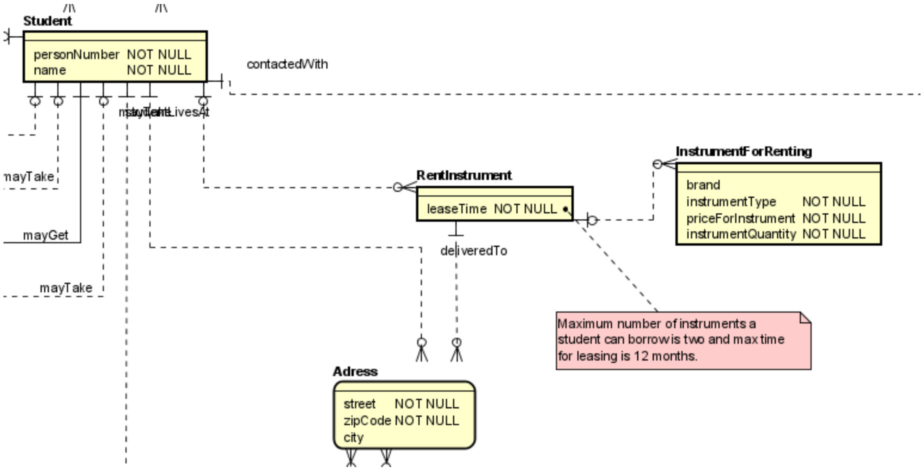
The solution for the payment and cost for the Student and Instructor is to split up the costs and payment for the different types of lessons. They all relate to the StudentMonthlyCost and InstuctorMonthlyPayment as we can see in figure 4. This way it's easier to change the payment for the different types of lessons.


PriceSingle
priceForSingleLessonStudent NOT NULL
paymentForInstructorSingle NOT NULL
PriceGroup
pricePerStudentGroup NOT NULL
paymentForInstructorGroup NOT NULL
PriceEnsemble
pricePerStudentEnsemble NOT NULL
paymentForInstructorEnsemble NOT NULL
StudentMonthlyCost
lessonsCost
instrumentRental
InstructorMonthlyPayment
indivdualLesson
groupLesson
ensembleLesson

The sibling discount may only affect the price the student pays, not the payment for the instructor.

Figure 4: The payment and cost solution where the relations of the payment goes to the student as well as the instructor.

The solution for the rental of instruments is made in a way so the instrument that is to be rented is separate from the main instrumentRental entity. This is to easier separate the data for the instruments and the renting process. There is also a comment added to explain a specific requirement from the text that is difficult to model in the conceptual model.



Maximum number of instruments a student can borrow is two and max time for leasing is 12 months.

Figure 5: Student renting instruments

The rest of the entities I won't show but they can be found in my GitHub(See introduction).

5 Discussion

I have read through the requirements multiple times so I would assume that everything that is required is in the CM, though there is a risk that something may have been missed.

One mistake that I felt that I have in my conceptual model is that it feels like there are too many relations, making the model feel cluttered and not comprehensive. One thing that I could do to fix that issue is to use inheritance. For example, instead of having three different relations to the three different type of lessons there could be a single relation to a super-class for those lessons. This would significantly improve the clarity of the conceptual model, while still gathering the same data.

There are attributes for every single entity, if there was an entity without attributes it would feel redundant to store that data.

The conceptual model uses the Lower Camel Case naming convention for the attributes and relations while classes(entities) use a Upper Camel Case.

All of the information that was difficulty to put into the CM as entities or attributes was made into comments to clarify the specific requirements.

6 Comments about course

In total it have taken probably around 20 hours of work total to study, make the CM and write this report.