Project Report - Task 1: Conceptual Model

Data Storage Paradigms, IV1351

November 6, 2024

Project members:
Nils Ekenberg, nilseke@kth.se
Samuel Engbom, sengbom@kth.se
Peter Li, pli2@kth.se

Declaration:

By submitting this assignment, it is hereby declared that all group members listed above have contributed to the solution. It is also declared that all project members fully understand all parts of the final solution and can explain it upon request.

It is furthermore declared that the solution below is a contribution by the project members only, and specifically that no part of the solution has been copied from any other source (except for lecture slides at the course IV1351), no part of the solution has been provided by someone not listed as a project member above, and no part of the solution has been generated by a system.

1 Introduction

The company Soundgood is a music school that sells music lessons to students. If a student wants to attend the school, an application is needed. There are four main subjects - lesson, student, instructor, payments. Each of the subjects include multiple different requirements.

Our task is to construct a Conceptual Model (CM) specified in the requirements and descriptions for Soundgood Music School. This project task is carried out as a collaboration between all three project members.

2 Literature Study

In order to carry out the task, the following literature and source of knowledge is used:

- Fundamentals of Database Systems (Seventh Edition) by Ramez Elmasri and Shamkant B. Navathe
- IV1351 Lectures and Canvas material such as the lecture slides and educational videos explaining Conceptual Modeling.

In the book, chapter three in particular, introduces and explains the modeling concepts of the Entity-Relationship (ER) model. This chapter also further explains CM modeling process, e.g. entity identification, attribute identification, relationship identification.

The lectures provide mostly a brief summary of content in the book to help further understanding the concepts. Furthermore, educational video materials in the Canvas page for Conceptual Model mainly explains the procedure to create a CM, as well as to introduce the basic usage for Astah, which is a tool to construct a CM.

3 Method

The CM is created as an ER diagram using the Astah software tool. For the design, procedures are followed that is outlined in the course material. Initially, a noun list is made, which consisted of all nouns present during the description of the project. Then, following the provided recorded course lectures, a category list is created in order to identify more potential candidates to become entities in the model. In the end, a discussion should be hold - regarding the entities present and expanded upon them in order to fill out the model.

After the diagram is populated with entities, upon further considerations, the unnecessary components in the model should be removed. This step involves renaming entities and changing into attributes if they were better suited as one.

Since every entity has to be described sufficiently by attributes, those that reflected the properties outlined in the requirements for the project are added. Making sure that the attributes describe an entity and are not suitable to be entities of their own right. Lastly, relationships that outline the interactions between them were created.

4 Result

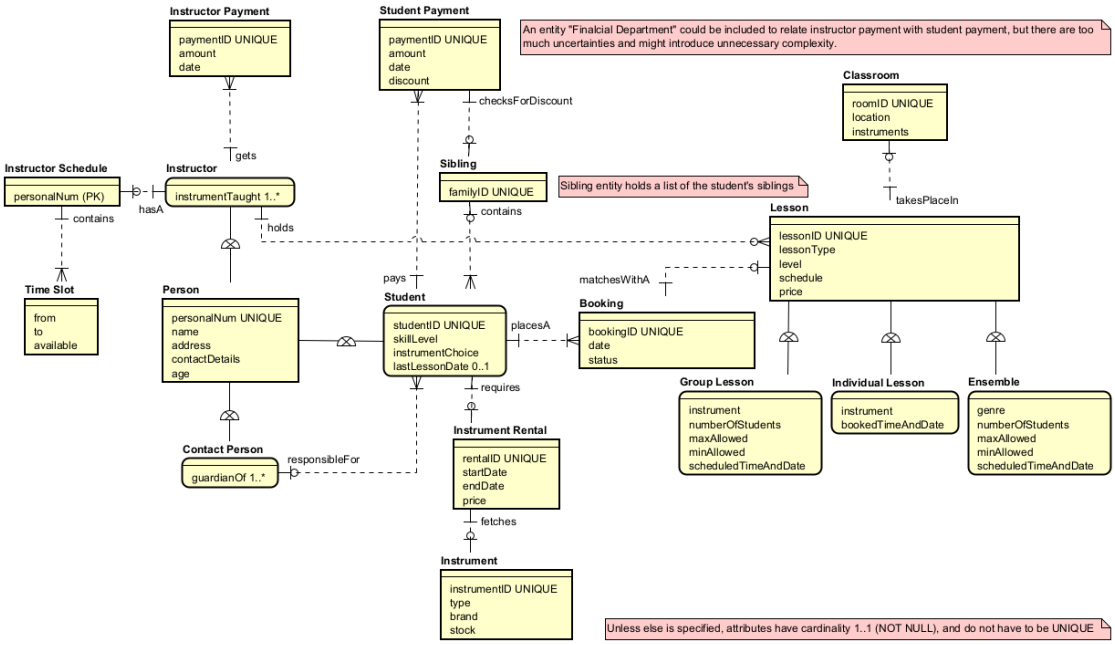

Figure 1: Conceptual Model for Soundgood music school.

The goal of the CM is to present all four main subjects, mentioned in the introduction - lesson, student, instructor, payments. As shown in Figure 1, all of the required subjects and their corresponding relationships are defined in the CM. Aside from the main subjects, a numerous of other necessary entities and attributes which were extracted from the task criteria are also present in each main subject.

In the finished CM, the cardinality for attributes is 1..1 unless something else is specified for a specific attribute. Similarly, an attribute cannot be NULL unless else is specified. This is because of the amount of attributes of this cardinality outnumber the other ones by a lot and as such makes the CM less cluttered thus a lot more readable.

It is assumed that one student can only rent one instrument. Since it is a rental, the `Instrument Rental` has a start and end date to it, as well as a price; and for storage purpose it needs a unique ID. When it comes to `Instrument`, there is a need to keep

track of the instrument type, stock, and possibly also its brand.

There are 3 types of lessons, `Group Lesson`, `Individual Lesson`, and `Ensemble`. They are subtypes of the entity `Lesson`. `Booking` can only match with one of these subtypes of `Lesson` per `Booking` instance.

The `Instructor Payment` and `Student Payment` entities are separate because the student discount could be useful to store. The `Student Payment` what the student pays and the `Instructor Payment` is what the instructor gets.

5 Discussion

The structure of the conceptual model was built around key entities such as the `Student`. In order to account for the functionality required, we modeled the enterprise around the requirements outlined in the project description.

In the model, a `Student` can rent instruments and book classes, different lesson types inherit properties of the lesson entity. Both `Student` and `Instructor` inherits attributes from the `Person` entity, ensuring redundancy, and keeps their own relevant attributes.

When it comes to the relation from instructors and students to the lesson, there is an argument to be made that they should point to the sub-entities of `Lesson`, however, this will make things cluttered and unsure if it is needed.

As mentioned in previous point, and in the results, reducing redundancies in the model is necessary, since it is key to ensure its readability and maintainability. Aside from inheritance, the majority of attributes had the necessary attribute (NOT NULL), hence a note that states the default setting of each attribute is NOT NULL. To improve readability, there are also challenges in the visual composition of the entities and relations. By improving the visual composition, it is much easier to understand how entities are related and if something is missing or incorrect in the CM.

The CM should contain all necessary entities and relations to operate. But since the current CM is more or less a first finished version, when the database is actually being built, it is certain that further adjustments for the model is needed to better adapt to the client's needs.

For instance, the task description mentioned a payment section where, supposedly, the instructor's payment has some correlation to how many lectures they have given during the month. It is not safe to assume that it is obvious to relate the student's payment with the instructor's payment, because there might not be sufficient data to model such relation in an appropriate way in this stage. An attempt was made to relate the two payments by creating an entity `Financial Department`, but it was unable to effectively

eliminate the uncertainty, and rather introduced more complexity to the model than necessary. In real world application, further communication with the client is needed to solve such uncertainties.