# Project Report

Data Storage Paradigms, IV1351

Murtadha Alobaidi

mhaao@kth.se

2021-06-09

# Project Report

# Sound good Music School

# Contents

# 1. Introduction

The report deals with and discusses the results from 4 tasks on IV1351 Data Storage Paradigms. It is about designing and discussing a program that can manage the database of a music school company. First task is about creating a conceptual model for the Sound good music school database. Second task translates the conceptual model to logical, physical models and creates database postages SQL or MySQL, and inserts data into a database. Third task creates OLAP, queries, and views in order to analyze the business. Fourth task develops part of Soundgood's website and contacts the database to JAVA using JDBC.

## 2. Literature Study

The teacher shared valuable resources on canvas that were base for this project. Additionally we read the book (ELMASRI, 2016). We saw as well Leif Lindbäck's videos that helped us to understand the purpose of designing a conceptual model and what it is important that should be kept in mind when it comes to creating a conceptual model for a database for first task. In order to complete second task we saw the video for logical and physical models. Additionally we read the book in order to get deeper understanding on ways of working with logical and physical and creation a database for them. The structured query language in the video by Paris Carbone helped us to understand the SQL to make the query. The video from Leif Lindbäck's introduction to JDBC showed the best way to connect the database to JAVA by using JDBC.

# 3. Method

## o Task 1 (Conceptual Model)

Before the group began with the first task, each group member read chapters 1-4 of the book (ELMASRI, 2016) and saw the video about how to Create a Domain Model, IE Notation and Conceptual Model. The goal of the first task was to create diagrams based on all data and the relation between Entity-Relation (ER). Firstly we took all (noun identification) Entities that could be relevant to "the Sound Good music school" database. We followed all steps according to the conceptual model video. To find more candidates and we used the Category list method. The group discussed which Entity-Relation (ER) might be relevant to our scenario and which should not be included according to the Conceptual model. The group discussed what attributes should be included and found the right relationships with them.

## o Task 2 (Logical and Physical Model)

The goal of the second task was to create Logical and physical models. We followed all the steps by the Leif Lindbäck video. Firstly we made the conceptual model and then we created a table for each entity in the conceptual model. Afterwards we created a column for each attribute that at most had one value. Attributes had cardinality zero or one. Next step was to create a new table for each column with a higher cardinality. The fourth step was to specify the type and domain for each column. Type had different characters for example char had specified length and varchar had no limits to the length and we could use these with string. The type helped when we insert data into the database using posters or MySQL. Next we considered column constraints if they were unique and different to null. That meant that the table had to have the same value for the column and not null means the cardinality was at least one and there was no higher cardinality. Then we assigned primary

keys(PK) to all strong entities. Strong entities have no total participation in the relation with other entities and can exist without any other entity. We assigned the primary to all strong entities. Later on we created a relation for all entities and used PK and the foregoing key. Then we cross-referenced the table for many-to-many relations. We did this by checking if we had atomic attributes. We had only a few tables with only composite primary keys and no other attributes. We had as well at least one table with two or more fields in addition to the primary key. Next we assigned FK to tables representing multi-valued attributes. PK of these tables was the FK with the multi-valued attribute combined. Then we verified if the model was normalized when it had only atomic attributes. In the last step it was possible to perform all planned operations. We performed all the planned actions as described in the description of the requirements.

- **Task 3 (SQL)**

In this task we worked with SQL language in order to create an OLAP, queries and views. The purpose of these queries was to analyze the business of the sound good music school. We used “SQL shell” terminal posters SQL to create a database. There were another way to make the database on pgAdmin4 but we preferred the SQL shell terminal to create it and to insert all data. During work with the queries we used Azure Data Studio because it’s easier to test the queries result during work. We used the visual Studio code to read and change the SQL data.

- **Task 4 (Programmatic Access)**

In this task, we used JDBC bank classes and changed the code so that it fitted the requested tasks in the project. What changed in the code were DAO, Controller, and BlockingInterpreter classes so that it included our requirements. This change was based on “Database Applications” which could be found on the Canvas page for the course. The

page contained as well a lecture video “Architecture and design of database application” by Leif Lindbäck.

## 4. Result

### ○ Task 1 (Conceptual Model)

Here we show where the conclusion from the conceptual model for the database in figure 1 Conceptual model IE diagram was created in the Astah program. Look at the link on https://github.com/mhaao/soundgood-music-school/blob/main/src/main/resources/IE.png.

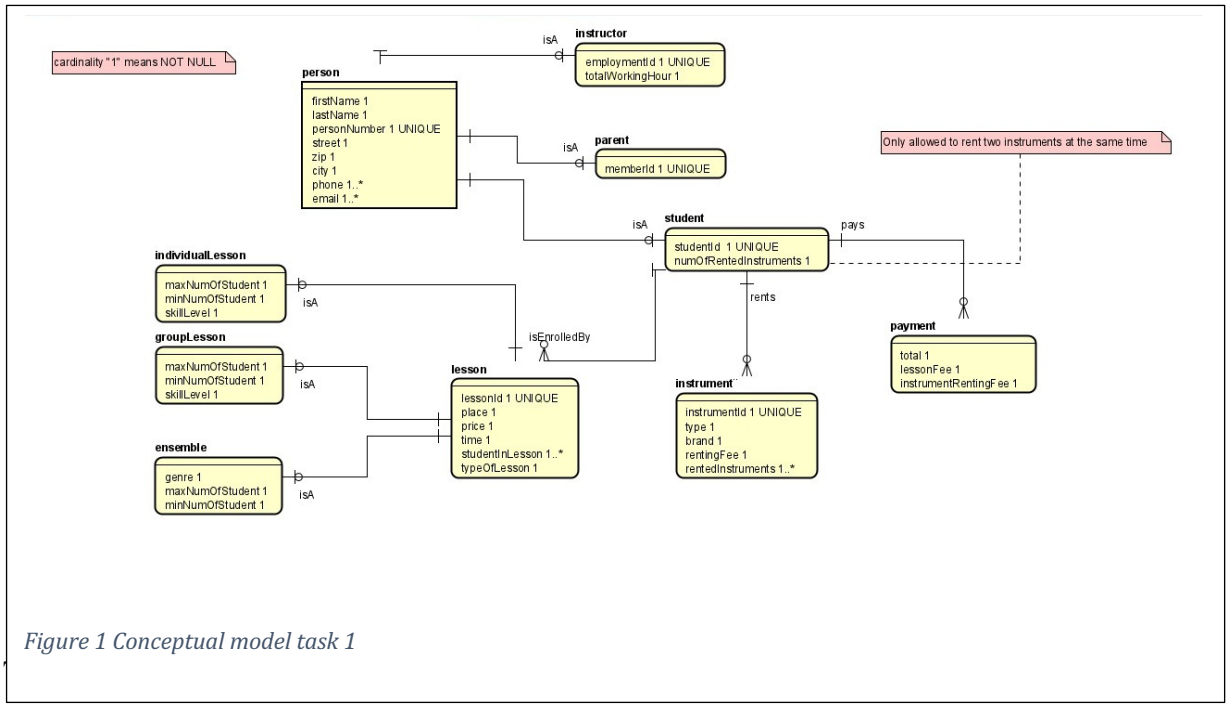


*Figure 1 Conceptual model task 1*

Figure 2. Physical models were created in the Atash program. Another part of this task was to create a database and insert all data. We used the posters SQL program to convert our

physical models from the Atash program to SQL. We needed to change a few things before we uploaded our data. We used the terminal to create the database and PgAdmin 4 in order to show all tables and all SQL queries that we created. We used as well posters SQL to insert all data about Sound good music school. We uploaded that to GitHub, ”Physical_model”https://github.com/mhaao/soundgood-musischool/blob/main/src/main/resources/physical_model.png , “database.sql” and “insert.sql” that you could find them here https://github.com/mhaao/soundgood-music-school/tree/main/src/main/resources.

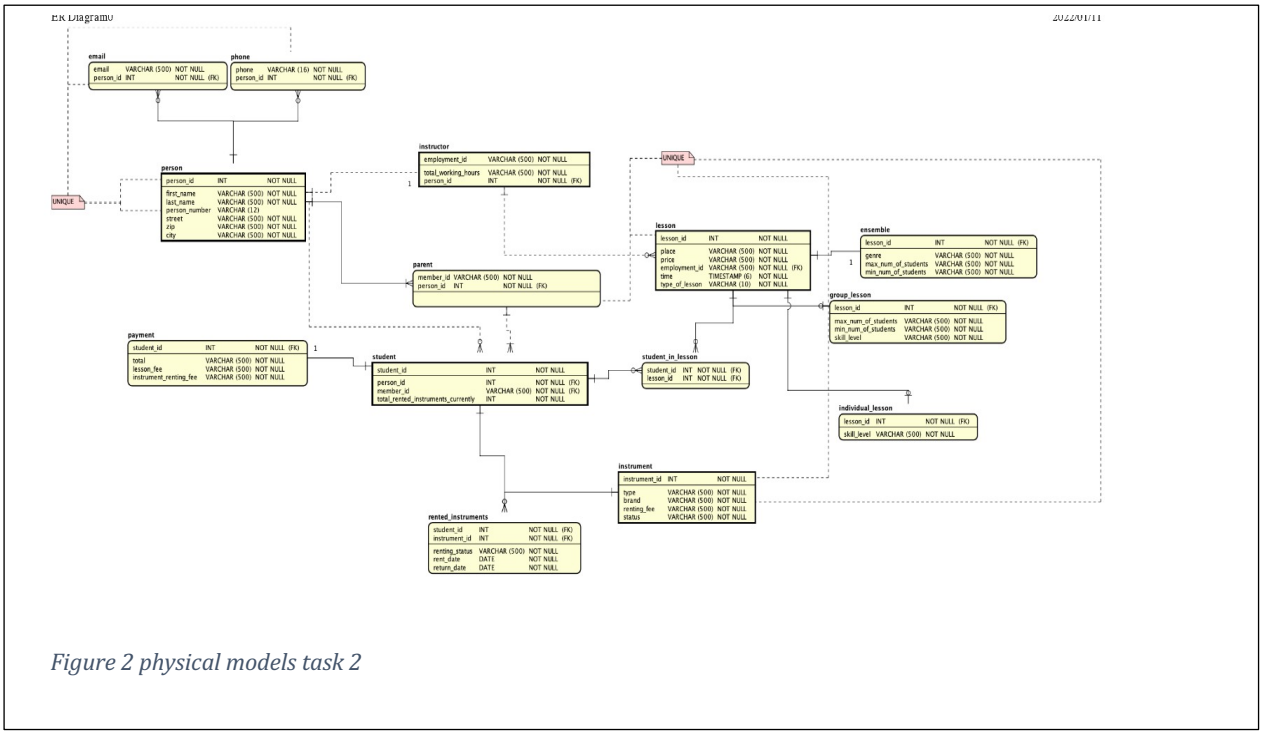


*Figure 2 physical models task 2*

- **Task 3 (SQL)**

The result of third task are available on a git repository and we had one SQL fil to all queries “view.sql”, all tables on “database.sql” and all insert data on “insert.sql”. You can find all this in the git repository [https://github.com/mhaao/soundgood-music-school/blob/main/src/main/resources/view.sql](https://github.com/mhaao/soundgood-music-school/blob/main/src/main/resources/view.sql) and on figure 3 showing the result av 4 queries.

Results Messages

| | month | year | ensemble | group | individual |
|---|---|---|---|---|---|
| 1 | 1 | 2021 | 0 | 0 | 2 |
| 2 | 5 | 2021 | 0 | 0 | 1 |
| 3 | 6 | 2021 | 0 | 0 | 1 |
| 4 | 7 | 2021 | 0 | 2 | 0 |
| 5 | 8 | 2021 | 0 | 0 | 2 |
| 6 | 9 | 2021 | 1 | 0 | 0 |

| | year | total_num_of_lessons | average_num_of_individual_per_month | average_num_of_group_per_month | average_num_of_ensemble_per_month |
|---|---|---|---|---|---|
| 1 | 2021 | 9 | 0.5000000000000000000 | 0.16666666666666666667 | 0.08333333333333333333 |

| | instructor_name | total_lessons | month | overtime |
|---|---|---|---|---|
| 1 | Maxie Maxie | 1 | 1 | False |
| 2 | Preston Preston | 1 | 1 | False |
| 3 | Preston Preston | 1 | 5 | False |
| 4 | Preston Preston | 1 | 6 | False |
| 5 | Preston Preston | 2 | 7 | True |
| 6 | Maxie Maxie | 2 | 8 | True |
| 7 | Preston Preston | 1 | 9 | False |

| | genre | max_num_of_students | min_num_of_students | type_of_lesson | year | week_number | day | status |
|---|---|---|---|---|---|---|---|---|
| 1 | Rock | 5 | 3 | Ensemble | 2022 | 3 | 11 | Full booked |

*Figure 3 all queries*

- **Task 4 (Programmatic Access)**

When the program was executed, we got a pad to write which command to execute. To see all commands we needed to type “help” and then we saw a list of all commands. It looked as in figure 4.


*Figure 4 all commands*

The command “list” indicate all information about an instrument in the school’s warehouse.


*Figure 5 list av all instrument*

The command “ensemble” showed all information about ensemble lessons coming following week. Note that this command update from today’s datum to one week.

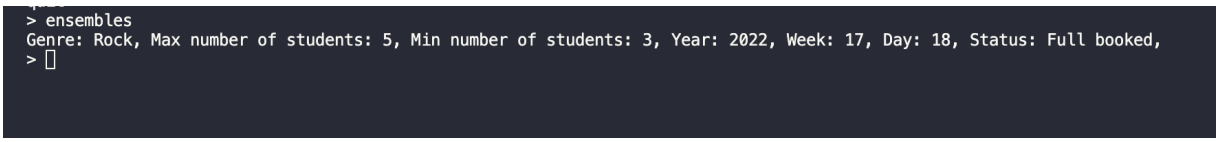


*Figure 6 list av all ensemble lessons*

In the command “rent” you needed to enter “student_id” and “instrument_id”. In case you entered the wrong ID, an exception was displayed without saving data in the database.

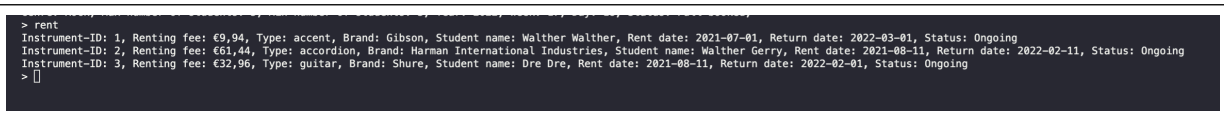


*Figure 7 register rented instrument*

The command “terminate” you need to enter “student_id” and instrument_id” to terminate ongoing rental. This data and all information about the rental saved in the database. If the data will not be updated in the database we use the rollback method to handle the Exception.

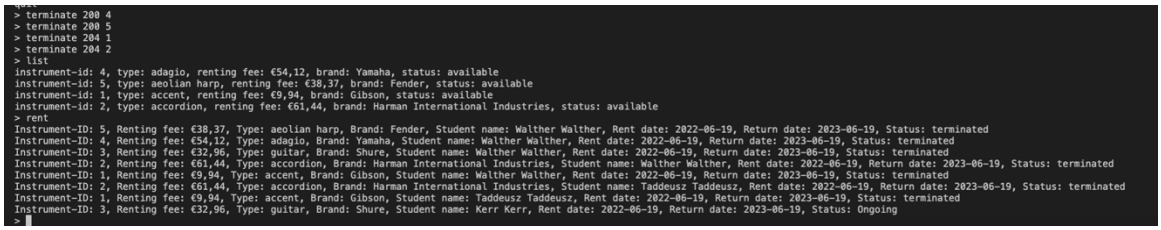


*Figure 8 register terminate instrument*

# 5. Discussion

- **Task 1 (Conceptual Model)**

We followed all naming conventions that were relevant to Soundgood Music School. We tried to avoid all errors that could be made with the help of the literature book and Lindbäck's video. We followed the method in "Fundamentals of database systems" of Ramez Elmasri to design errorfree, easy to understand and follow diagrams. Yes, we followed all the steps with the IE diagram and we managed to make a diagram. We had all the important entities that helped with the project. The diagram contained all entities that were visible. We used all relevant relations and we made the design flexible in order to make it easier to work with for example if we needed to change something in the design.

- **Task 2 (Logical and Physical Model)**

All the naming conventions that were related to our project were followed and explained. We followed all the crowfoot notation correctly. Our physical models were in 3NF and we worked by introductions in the video recorded by Leif Lindbäck about physical and logical models. Column constraints and foreign keys were specified and could be used without a doubt. We chose carefully all primary keys so they were motivated in a good way. The relationships in our model met our original requirements from the project requirements description.

- **Task 3 (SQL)**

We used to query one of three tables together with FULL OUTER JOIN. We took all columns we needed from different tables that had the same attribute. EXTRACT to took the year and month we needed from the timetable. We used CASE and WHEN to take the right terms and by using, GROUP BY to show these columns. Query 2 used FULL OUTER JOIN also to take all columns we needed from different tables and by COUNT we took the

average number for each lesson (individual, group, and ensemble). Query 3 used INNER JOIN to take all columns that matched each other. AND merged columns one with name and other with surname in order to get all full names of students. GROUP BY showed all chosen columns. Query 4 we used a keyword similar to other queries. Here however we used the NOW day and NOW to get next week and NOW year. These keywords helped to take days' data. We used all these keywords to extract certain information from various columns. We used Azure Data Studio to test the results.

- **Task 4 (Programmatic Access)**

In this task, the requirements were to develop the website for Soundgood Music School. The goal was to ensemble lessons coming following week, to be able to rent instruments from the school and to end the rent for instruments. We developed the command interface that was linked to the website. All requirements for the task were made. The program compared with and updated new information existing database. The school always had available up to date overview of their instruments status or lessons status. The program worked as expected.