**CCT College Dublin**

**Assessment Cover Page**

|  |  |
| --- | --- |
| **Module Title:** | Databases |
| **Assessment Title:** | ERD |
| **Lecturer Name:** | Aldana Louzan |
| **Student Full Name:** | Halil UGUR |
| **Student Number:** | 2022389 |
| **Assessment Due Date:** | 28/10/2022 @23:59 |
| **Date of Submission:** | 23.10.2022 |

**Declaration**

|  |
| --- |
| By submitting this assessment, I confirm that I have read the CCT policy on Academic Misconduct and understand the implications of submitting work that is not my own or does not appropriately reference material taken from a third party or other source. I declare it to be my own work and that all material from third parties has been appropriately referenced. I further confirm that this work has not previously been submitted for assessment by myself or someone else in CCT College Dublin or any other higher education institution. |

Contents

[Abstract 3](#_Toc117535925)

[Conceptual Design (CHEN Notation) 4](#_Toc117535926)

[Logical Design (Crow's Foot Notation) 5](#_Toc117535927)

[Normalization process 6](#_Toc117535928)

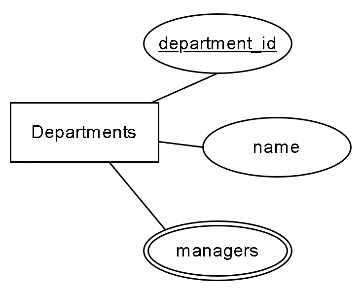
[References 8](#_Toc117535929)

# Abstract

This project has aim that create database for Human Resources. Project generally has three sections. These are conceptual design, logical design, and normalization process. The conceptual design contains generally structure of the HR database. The logical design gives us more detail about the HR database. The normalization process is trying to block of data repeating or null values.

# Conceptual Design (CHEN Notation)

There is have two main entities for conceptual design as Employee and Department. Employee entity include that personal information. Department entity is any field work on. Each entity has relationship that as shown below.

Diagram of Employee


FINAL RESULT

Diagram

Description automatically generated

# Logical Design (Crow's Foot Notation)

At the logical design section created relational data model from conceptual design. On RDM added relations to between entities.

Diagram

Description automatically generated

The above image have simple relation between employees and departments. There are several reasons why I draw like this:

1. Each employee can work different time at the different departments. We need to create a relationship table for periods.
2. Employee can be manager at the same time. So, we created the manage relation between employee and department
3. Employee should have old salary data on theirself. For that reason I added salary and salary\_date on employee table.

Table

Description automatically generated

Figure 1 Before Normalization process table visuals

# Normalization process

After I filled data, I saw that we have repeat data on employee table. At the same time, we needed to split composite fields to subs fields. For this I added new tables on current schema.

Steps of Normalization process:

* Employee table was had duplicate data. Therefore, we marked the repeat values on table.
* We created Salaries and Jobs table because same data always repeated.
* Salary history should have different table without contract table. Because salary can a few times increment in contract periods. So, it needs to independent from contracts table. Job history table too.
* We need divide each field that require to sub fields. Composite fields always contain minimum two values. Therefore, this makes difficult understanding of the data and query.

Diagram, schematic

Description automatically generated

Table

Description automatically generated

Figure 2 After Normalization process Table visuals

# GitHub Repository

All the materials we use can be accessed from this repo: [GitHub Repo](https://github.com/halilugur/database_ca1)

# References

1. MySQL (2022a). MySQL :: MySQL Workbench Manual :: 9 Database Design and Modeling. [online] dev.mysql.com. Available at: https://dev.mysql.com/doc/workbench/en/wb-data-modeling.html.
2. MySQL (2022b). MySQL :: MySQL Workbench Manual :: 9.4.2.2 Reverse Engineering a Live Database. [online] dev.mysql.com. Available at: https://dev.mysql.com/doc/workbench/en/wb-reverse-engineer-live.html [Accessed 23 Oct. 2022].
3. Wikipedia Contributors (2019a). Conceptual Schema. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/Conceptual\_data\_model [Accessed 5 Jan. 2020].
4. Wikipedia Contributors (2019b). Entity Relationship Model. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/Entity%E2%80%93relationship\_model.
5. Wikipedia Contributors (2021). Logical Schema. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/Logical\_data\_model [Accessed 23 Oct. 2022].
6. Wikipedia Contributors (2022). Physical Schema. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/Physical\_data\_model [Accessed 23 Oct. 2022].