USE CASE PROBLEM DEFINITION

<u>**Group Number**</u> – 3

<u>**Team Members:**</u>   Mayur Mahanta   Moheth Muralidharan

EMPLOYEE RATING APPLICATION

<u>**Business Problem:**</u>

A Manufacturing Company is planning to develop a new database system which will support its new employee rating application. The database is designed particularly to decrease biasedness and increase transparency across the company. The rating of an employee will be done on a pre-defined set of scores that will be utilized to determine their individual input percentage and efficiency of their work deliverance. The rating will be done by a group of reviewers for all the employees based on their onsite performance. The final score will depend on multiple ratings and keeping in mind the transparency factor, may be/will be viewed by everyone in the company.

There are a standard set of rules that needs to be followed for the designed model for the above specified application. They are as follows -

- Each employee will need to finish their work.
- The employee will have one designation. Under certain conditions an employee can have more than one designation.
- Every employee will belong to a department.
- The employee will be rated by the group of reviewers with scores.
- All work will be rated by reviewers individually.
- The reviewer can post the scores for every work he participated in.
- The scores can be viewed by the employees.

<u>**Design:**</u>

The design constraints or the entities for the above Employee Rating Application System are as follows –

- Employee
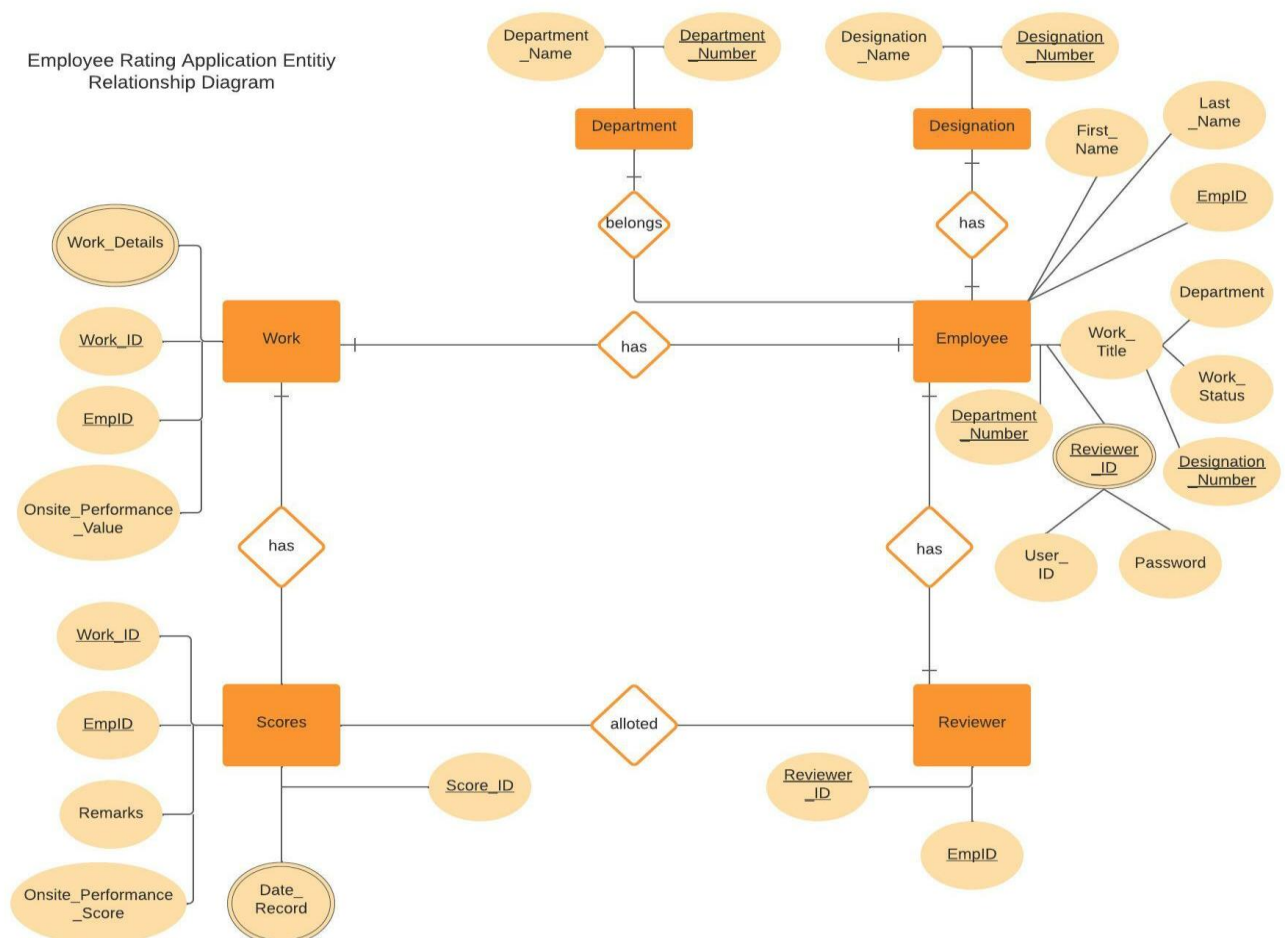- Designation
- Work
- Department
- Scores
- Reviewer

The above given design constraints or entities will generate the tables in our database according to the design and implementation of the database schema. These tables are formed to meet the requirements and the standard set of rules defined above for our database. Firstly, the EMPLOYEE table will contain all concerned details along with its designation and department. Secondly, the DESIGNATION and DEPARTMENT tables will have a set of details which are required to review and distinguish by the reviewers while scoring. However, these set of details will not have an impact on determining the onsite performance in order to keep the transparency and unbiasedness factor intact across the company. Next, all the employees/workers will have been assigned work and the

evaluation of their onsite performance will be done on the basis of these works. For the evaluation to take place for each individual employee and be added for scoring, they will be required to finish their work. Thirdly the WORK table will also contain a set of details relevant to the table which will be later used by the reviewers while evaluating and scoring each individual employee. Again, the REVIEWER table will comprise of information related to the reviewer. There will be a group of reviewers who will be rating and scoring employee/work. Reviewer will rate the employee/work by allotting scores. Lastly, the SCORES table will give the ratings and evaluate each employee's onsite-performance. As specified above under the standard set of rules, the scores may/can be accessed/viewed by each employee in the company.
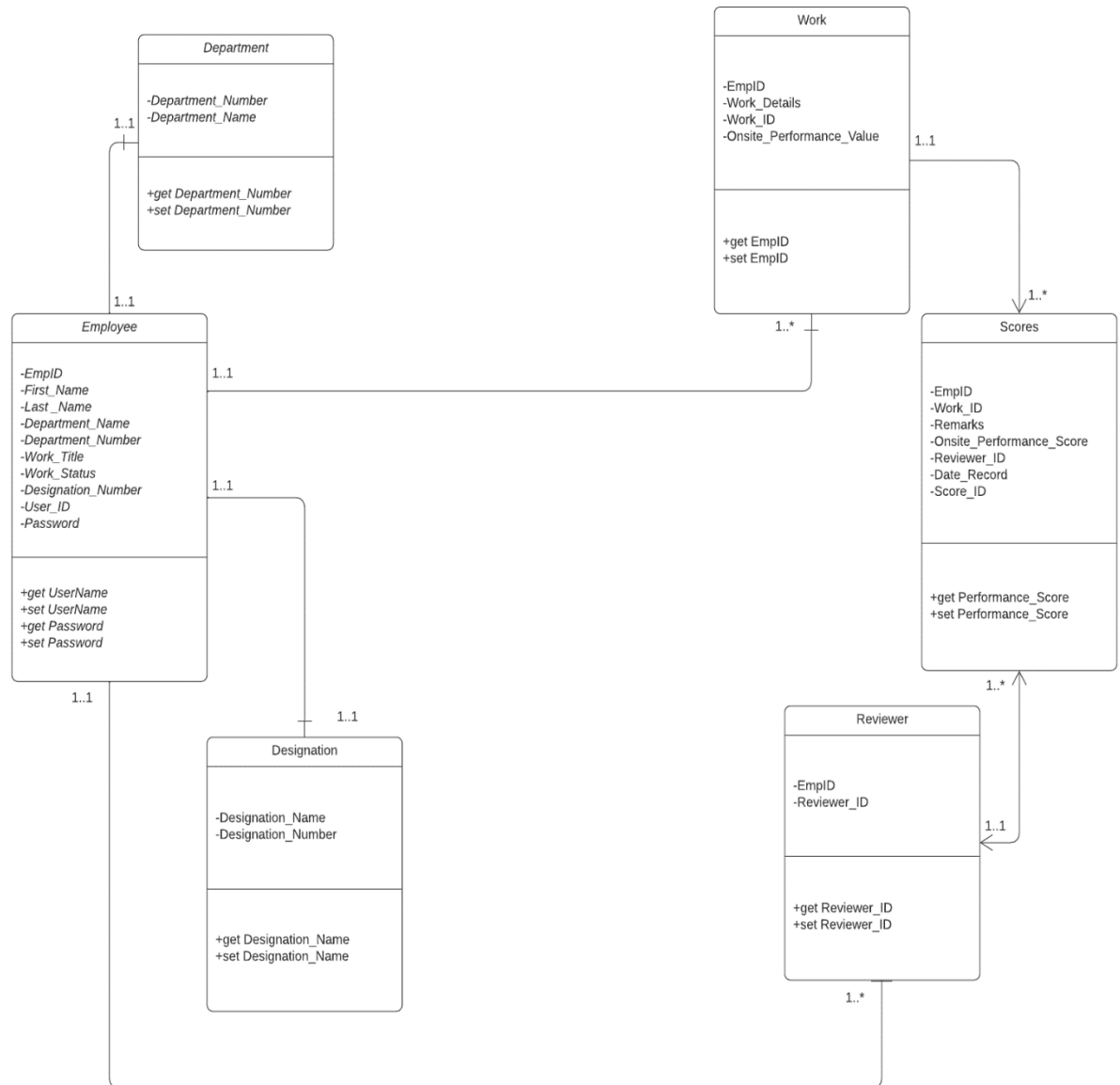
Now, the attributes for the above listed Entities are given below –

- **Employee** (EmpID, First_Name, Last_Name, User_ID, Password, Designation_Name, Designation_Number, Department_Name, Department_Number, Work_Title, Work_Status, Reviewer_ID)
- **Designation** (Designation_Name, Designation_Number)
- **Work** (Work_ID, Work_Details, EmpID, Onsite_Performance_Value)
- **Department** (Department_Name, Department_Number)
- **Scores** (Score_ID, EmpID, Work_ID, Reviewer_ID, Onsite_Performance_Score, Remarks, Date_Record)
- **Reviewer** (Reviewer_ID, EmpID)

Entity Relationship Diagram:

Unified Modelling Language (UML) Class Diagram:

Relational Model:

Employee (EmpID, First_Name, Last_Name, User_ID, Password, Designation_Name, Designation_Number, Department_Name, Department_Number, Work_Title, Work_Status, Reviewer_ID)

       Employee: EmpID in relation to Employee: NULL not allowed, on delete/update cascade.

Designation (Designation_Name, Designation_Number)

       Here, Designation_Number is the primary key.

Department (Department_Name, <u>Department_Number</u>)

      Here, Department_Number is the primary key.

Work (<u>Work_ID</u>, Work_Details, <u>EmpID</u>, Onsite_Performance_Value)

      Here, Work_ID is the primary key.

      <u>Employee: EmpID in relation to Employee:</u> NULL not allowed, on delete/update cascade.

Reviewer (<u>Reviewer_ID</u>, <u>EmpID</u>)

      Here, Reviewer_ID is the primary key.

      <u>Employee: EmpID in relation to Employee:</u> NULL not allowed, on delete/update cascade.

Scores (<u>Score_ID</u>, <u>EmpID</u>, <u>Work_ID</u>, <u>Reviewer_ID</u>, Onsite_Performance_Score, Remarks, Date_Record)

      Here, Score_ID is the primary key.

      <u>Employee: EmpID in relation to Employee:</u> NULL not allowed, on delete/update cascade.
      <u>Reviewer: Reviewer_ID in relation to Reviewer:</u> NULL not allowed, on delete/update cascade
      <u>Work: Work_ID in relation to Work:</u> NULL not allowed, on delete/update cascade.