MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL
*(A constituent unit of MAHE, Manipal)*

# Mini Project Report

of

# Database Systems Lab (CSE 2262)

## Factory Management System

SUBMITTED
BY


Arnav Gupta | RegNo.210905348 | RollNo.55 | CSE-D

Ditya Chawla | RegNo.210905266| RollNo.41 | CSE-D



Department of Computer Science and Engineering Manipal Institute of Technology, Manipal.


April 2023

MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL
*(A constituent unit of MAHE, Manipal)*
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Manipal
13/05/2023

# CERTIFICATE

This is to certify that the project titled Factory Management System is a record of the bonafide work done by Arnav Gupta and Ditya Chawla submitted in partial fulfilment of the requirements for the award of the Degree of Bachelor of Technology (B.Tech.) in COMPUTER SCIENCE & ENGINEERING of Manipal Institute of Technology, Manipal, Karnataka, (A Constituent Institute of Manipal Academy of Higher Education), during the academic year 2022-2023.

## Name and Signature of Examiners:

1. Dr. Roopalakshmi R, Associate Professor, CSE Dept.

2. Prof. Tanuja Shailesh, Assistant Professor, CSE Dept.

# TABLE OF CONTENTS

# ABSTRACT

This project focuses on developing a factory management system based on MYSQL, which includes an employee table, an attendance table, department, and project details. The system will allow managers to track employee attendance, assign tasks, and monitor project progress. The employee table will include information such as name, Date, and department, while the attendance department will record employee attendance. The project details module will track the status of ongoing projects. The system will provide a comprehensive view of the factory's operations, enabling managers to make informed decisions and optimize performance. The project will be developed using SQL programming and database management tools. Overall, this system will improve the efficiency of factory management, enhance productivity, and reduce costs.

# INTRODUCTION

The factory management system is an essential tool for any manufacturing facility. It helps managers to streamline the processes and optimize the performance of the factory. In today's digital age, the use of technology in the workplace has become more prevalent, and factory management systems based on SQL are one such example. This project aims to develop a factory management system based on SQL, which will include an employee table, an attendance department, and project details. The system will provide managers with a comprehensive view of the factory's operations and enable them to make informed decisions. This project will be developed using SQL programming and database management tools. The final system will improve the efficiency of factory management, enhance productivity, and reduce costs. The following sections will outline the detailed features of the system and the methodology used in developing it.

The user interface is designed to be simple and intuitive, allowing users to navigate the platform and check whats currently happening in the factory. The platform will also provide a clear display of the attendance of employee, their details,also the projects they are working on The project aims to provide an efficient management system for small scale factory Owners.

## PROBLEM STATEMENT

The manufacturing industry faces several challenges in optimizing their operations and increasing productivity. One significant challenge is the management of human resources, including tracking employee attendance and project details. Traditional methods of managing these tasks can be time-consuming, prone to errors, and require significant administrative effort. Additionally, without a comprehensive view of project status and employee performance, managers may struggle to identify potential bottlenecks and make informed decisions to optimize performance.

Data Requirements:

- Employee Registration: Employee Must be registered in the database and be assigned details like Employee_id, Assigned Department_id,also his/her first and last name.
- Department List: There has to be a department list already in the database.
- Live Projects: There should be some live projects listed under the project table

# METHODOLOGY

The methodology used in developing the Factory management system with a follows a structured approach that involves two stages. The stages include design and implementation.

Design:
During the design stage, Entity Relationship (ER) diagrams are created to represent the relationships between the different entities in the system. These diagrams are used to design the database schema and ensure the system can accurately store and retrieve data. The ER diagrams are also useful in identifying potential areas for optimization.

Relational tables are used to store Employee Data, Departments,Live projects,Attendance,and the education background of certain employees. These tables are designed to minimize data redundancy and ensure data integrity. Industry standard tools such as SQL Plus are used to design and create the database schema.

In addition to the database design, creating a user-friendly UI/UX is also important. The user interface is designed to be intuitive and easy to use. Python Tkinter was used for the front-end to create an easy to use application.

Implementation:
For the implementation stage, Oracle DB was used as database management system. Oracle DB is a robust, scalable, and secure system that can handle large volumes of data.

# ER DIAGRAM

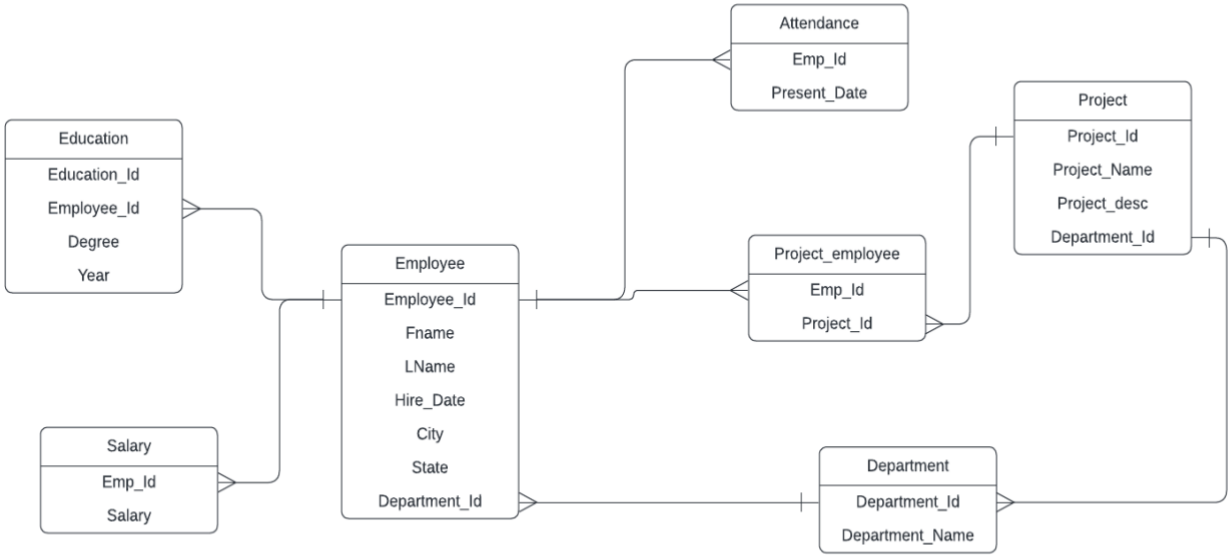
The Employee table has a one-to-many relationship with the Project table, as many employees can work on a project.

The project table has one to many relationship with department as there are many projects that may come under different departments.

The Education table has one to one relationship as one employee can have on record of their education.

The salary table has one to one relationship with employee table as each employee can have only salary

The department table has one to many Relationship with Employee table as there can be more than one employee in one department.


Attendance
Emp_Id
Present_Date
Education
Education_Id
Employee_Id
Degree
Year
Project
Project_Id
Project_Name
Project_desc
Department_Id
Employee
Employee_Id
Fname
LName
Hire_Date
City
State
Department_Id
Project_employee
Emp_Id
Project_Id
Salary
Emp_Id
Salary
Department
Department_Id
Department_Name

# DDL COMMANDS TO CREATE TABLE WITH INTEGRITY CONSTRAINTS

```
CREATE TABLE Employee(
  Employee_Id int PRIMARY Key,
  First_Name VARCHAR(25),
  Last_Name VARCHAR(25),
  Hire_Date DATE,
  City VARCHAR(25),
  State VARCHAR(25),
Department_Id int,
Foreign key(Department_Id) references department(Department_Id)
  );

CREATE TABLE Department(
  Department_Id int primary Key,
  Department_Name VARCHAR(30)
  );

CREATE TABLE Project(
  Project_Id int primary key,
  Project_Name VARCHAR(50),
  Project_Description VARCHAR(50),
Departement_Id int
Foreign key(Department_Id) references department(Department_Id)
  );


CREATE TABLE Education(
  Education_Id int primary key,
  Employee_Id int,
  Degree VARCHAR(30),
  Graduation_Year int(4),

  FOREIGN KEY (Employee_Id)
      REFERENCES Employee(Employee_Id)
  );
```

```
create table salary
(emp_id int,salary int,
Foreign key(emp_id) references Employee(Employee_Id)
);

CREATE TABLE Attendance(
  Emp_Id int,
  Preday int,
Foreign key(emp_id) references Employee(Employee_Id)
  );

Create table projectemp(
Emp_Id int,
Proj_Id int,
Foreign key(emp_id) references Employee(Employee_Id)
);
```

Query1: Displaying members of a team.
select * from players where Team_Name = 'MIT-A';

Query2: Display all the goalkeepers select
* from players where position = 'GK';

Query3: Players from MIT C who have scored more than one goal
select Name,goals from players where goals > 0 and pid in (select pid from players where team_name = 'MIT-C');

Query4: Display striker with least goals among other strikers select Name,goals from players where position = 'Striker' and goals <= all(select goals from players where position = 'Striker');

## Some Snapshots:

| Employee_Id | First_Name | Last_Name | Hire_Date | City | State | Department_id |
|---|---|---|---|---|---|---|
| 101 | Ojas | Phansekar | 2016-04-14 | Hyderabad | Andhra Pradesh | 1 |
| 102 | Vrushali | Patil | 2018-06-07 | Itanagar | Arunachal | 2 |
| 103 | Pratik | Parija | 2019-09-06 | Dispur | Assam | 3 |
| 104 | Chetan | Mistry | 2011-10-13 | Patna | Bihar | 4 |
| 105 | Anugraha | Varkey | 2017-11-10 | Purnia | Bihar | 5 |
| 106 | Rasagnya | Patel | 2015-10-29 | New Delhi | Delhi | 2 |
| 107 | Pratibha | Goel | 2016-09-15 | Kanpur | Uttar Pradesh | 1 |
| 108 | Tanvi | Sahni | 2017-05-21 | Noida | Uttar Pradesh | 3 |
| 109 | Anit | Ram | 2018-09-29 | Manipal | Karnataka | 4 |
| 110 | Dayita | Khosla | 2011-04-12 | Panaji | Goa | 1 |
| 111 | Vishal | Parmar | 2020-11-12 | Ranchi | Jharkhand | 5 |
| 112 | Darshana | Kara | 2016-10-23 | Kerala | Thiruvananthapuram | 5 |
| 113 | Arnav | Balasubramanian | 2017-04-16 | Hyderabad | Andhra Pradesh | 4 |
| 114 | Isha | Kale | 2018-06-15 | Itanagar | Arunachal | 1 |
| 115 | Pallav | Choudhry | 2019-08-12 | Dispur | Assam | 2 |
| 116 | Karan | Bal | 2011-07-12 | Patna | Bihar | 3 |
| 117 | Pratap | Banerjee | 2016-11-28 | Purnia | Bihar | 4 |
| 118 | Sai | Rana | 2021-06-30 | New Delhi | Delhi | 5 |
| 119 | Darshana | Deshpande | 2020-04-04 | Kanpur | Uttar Pradesh | 1 |
| 120 | Devi | Dhar | 2018-01-22 | Noida | Uttar Pradesh | 2 |
| 121 | Anand | Char | 2017-02-13 | Manipal | Karnataka | 3 |
| 122 | Chetan | Mistry | 2019-03-12 | Panaji | Goa | 4 |
| 123 | Anugraha | Varkey | 2015-12-15 | Ranchi | Jharkhand | 5 |
| 124 | Rasagnya | Patel | 2014-05-20 | Kerala | Thiruvananthapuram | 2 |

```
mysql> select * from department;
+---------------+-----------------------+
| Department_Id | Department_Name       |
+---------------+-----------------------+
|             1 | Human Resources       |
|             2 | Software Development  |
|             3 | Data Analysis         |
|             4 | Data Science          |
|             5 | Business Intelligence |
|             6 | Data Engineering      |
|             7 | Manufacturing         |
|             8 | Quality Control       |
+---------------+-----------------------+
```

```
mysql> select * from project;
+------------+--------------+--------------------+---------------+
| Project_Id | Project_Name | Project_Description | Department_id |
+------------+--------------+--------------------+---------------+
|         21 | Dev          | Whatever           |             1 |
|         22 | Prod         | do something       |             2 |
|         23 | Test         | focus              |             2 |
|         24 | Nothing      | do nothing         |             3 |
|         25 | Research     | focus on everything |             4 |
|         26 | Next Steps   | find some way out  |             5 |
+------------+--------------+--------------------+---------------+
```

```
mysql> select * from education;
+--------------+-------------+--------+-----------------+
| Education_Id | Employee_Id | Degree | Graduation_Year |
+--------------+-------------+--------+-----------------+
|           10 |         101 | BA     |            2017 |
|           11 |         102 | BA     |            2019 |
|           12 |         104 | BA     |            2011 |
|           13 |         108 | BA     |            2015 |
|           14 |         109 | BBA    |            2013 |
|           15 |         107 | BBA    |            2008 |
|           16 |         106 | BBA    |            2007 |
+--------------+-------------+--------+-----------------+
```

```
mysql> select * from salary;
+--------+--------+
| emp_id | salary |
+--------+--------+
|    101 |  10000 |
|    102 |  15000 |
|    103 |  15000 |
|    104 |  17000 |
|    105 |  18000 |
|    106 |  18500 |
|    107 |  19000 |
|    108 |  19000 |
|    109 |  19500 |
|    110 |  12000 |
|    111 |  22000 |
|    112 |  25000 |
|    113 |  23000 |
|    114 |  24000 |
|    115 |  14000 |
|    116 |  21000 |
|    117 |  23000 |
|    118 |  29000 |
|    119 |  18000 |
|    120 |  19000 |
|    121 |  35000 |
|    122 |  22000 |
|    123 |  32000 |
|    124 |  34000 |
+--------+--------+
```

```
+--------+---------+
| Emp_id | proj_id |
+--------+---------+
|    101 |     121 |
|    101 |     121 |
|    102 |     121 |
|    103 |     121 |
|    104 |     121 |
|    105 |     121 |
|    106 |     121 |
|    107 |     121 |
|    108 |     121 |
|    109 |     122 |
|    110 |     122 |
|    111 |     122 |
|    112 |     122 |
|    113 |     122 |
|    114 |     122 |
|    115 |     122 |
|    116 |     122 |
|    117 |     123 |
|    118 |     123 |
|    119 |     123 |
|    120 |     123 |
|    121 |     123 |
|    122 |     123 |
|    123 |     123 |
|    124 |     123 |
|    101 |     124 |
|    103 |     124 |
|    105 |     124 |
|    107 |     124 |
|    109 |     124 |
|    111 |     124 |
|    113 |     124 |
|    115 |     124 |
|    117 |     125 |
|    119 |     125 |
|    121 |     125 |
|    123 |     125 |
|    124 |     125 |
|    122 |     125 |
|    108 |     125 |
|    106 |     125 |
|    108 |     126 |
|    106 |     126 |
|    104 |     126 |
|    110 |     126 |
|    112 |     126 |
|    114 |     126 |
+--------+---------+
```

```
mysql> select Emp_Id,Project_Id,Project_Name from projectemp,project where emp_id=101;
+--------+------------+--------------+
| Emp_Id | Project_Id | Project_Name |
+--------+------------+--------------+
|    101 |         21 | Dev          |
|    101 |         21 | Dev          |
|    101 |         21 | Dev          |
|    101 |         22 | Prod         |
|    101 |         22 | Prod         |
|    101 |         22 | Prod         |
|    101 |         23 | Test         |
|    101 |         23 | Test         |
|    101 |         23 | Test         |
|    101 |         24 | Nothing      |
|    101 |         24 | Nothing      |
|    101 |         24 | Nothing      |
|    101 |         25 | Research     |
|    101 |         25 | Research     |
|    101 |         25 | Research     |
|    101 |         26 | Next Steps   |
|    101 |         26 | Next Steps   |
|    101 |         26 | Next Steps   |
+--------+------------+--------------+
18 rows in set (0.01 sec)
```

```
mysql> select count(Emp_id) from attendance where emp_id=103;
+---------------+
| count(Emp_id) |
+---------------+
|            11 |
+---------------+
1 row in set (0.01 sec)
```

PL/SQL QUERY:

```
mysql> SELECT d.Department_Name, COUNT(e.Employee_Id) AS Employee_Count, AVG(s.salary) AS Average_Salary
    -> FROM department d
    -> JOIN employee e ON d.Department_Id = e.Department_Id
    -> JOIN salary s ON e.Employee_Id = s.emp_id
    -> GROUP BY d.Department_Name
    -> HAVING AVG(s.salary) > 20000;
+----------------------+----------------+----------------+
| Department_Name      | Employee_Count | Average_Salary |
+----------------------+----------------+----------------+
| Software Development |              5 |     22110.0000 |
| Data Analysis        |              4 |     24750.0000 |
| Data Science         |              5 |     22990.0000 |
| Business Intelligence |              5 |     27720.0000 |
+----------------------+----------------+----------------+
4 rows in set (0.00 sec)
```

```
mysql> UPDATE salary
    -> SET salary = salary * 1.1;
Query OK, 24 rows affected (0.04 sec)
Rows matched: 24  Changed: 24  Warnings: 0

mysql> UPDATE salary s
    -> SET s.salary = s.salary * 1.1
    -> WHERE s.emp_id IN (
    ->   SELECT a.emp_id
    ->   FROM attendance a
    ->   WHERE a.predate >= DATE_SUB(NOW(), INTERVAL 7 DAY)
    ->   GROUP BY a.emp_id
    ->   HAVING COUNT(*) > 7
    -> );
```

# LIMITATIONS AND FUTURE WORK

## LIMITATIONS:

1. Limited scalability: The system is designed for small to medium-sized factories, and as the size of the factory grows, it may not be able to handle the increased volume of data and user traffic.
2. Lack of integration with other systems: The system is a standalone application and does not integrate with other enterprise systems such as ERP, CRM, or SCM, which could limit its usefulness in larger factories.
3. User interface limitations: While the system's user interface is intuitive and easy to use, it may not meet the specific requirements of some users, such as those with accessibility needs or users who require a more customizable interface.
4. Limited functionality: The system focuses on employee management, attendance tracking, and project details, and it may not meet all the requirements of more complex factories with specific needs, such as inventory management, production planning, or quality control.

Overall, while the factory management system developed using Tkinter as the front-end is a useful tool for small to medium-sized factories, it may have limitations that need to be considered before implementation.

## FUTURE WORK:

Automating all employee-based tasks, including attendance management, project assignment, progress monitoring, and salary calculations based on overtime work, would streamline the entire process and reduce the workload on managers and HR personnel. By implementing this system, the factory can ensure that all tasks are completed efficiently and accurately, leading to increased productivity and employee satisfaction.

Additionally, integrating functionality for employee performance evaluations and tracking progress towards career goals can help identify high-performing employees and provide opportunities for growth and development. This can lead to increased employee engagement and retention.

Overall, automating all employee-based tasks and providing a comprehensive solution for factory management can lead to improved efficiency, accuracy, and employee satisfaction, ultimately contributing to the success of the factory.

# CONCLUSION

The leader board has very versatile features to use for a big tournament with further improvement required and it would be of great use to us when we organize tournaments.

During the design stage, we used Entity Relationship diagrams to create the database schema, and relational tables were designed to ensure accurate data storage and retrieval. These design decisions allow for efficient management of the user data and an intuitive ranking system to showcase player progress.

In the implementation stage, the project was developed using the reliable and flexible combination of Oracle DB. This technology stack allows for seamless data processing, fast load times, and easy scalability.

While the project has some limitations, such as the lack of a password reset feature and the potential for cheating, several future work possibilities exist to address these issues and further improve the platform. Future work could include implementing anti-cheat measures, expanding the game to include additional features, and improving the user interface with advanced design elements.

# References


- Nodejs application with oracle dB

https://www.oracle.com/database/technologies/appdev/quickstartnodeonprem. html

- Mysql implementation: https://www.javatpoint.com/mysql- queries
- Tool for creating ER Diagram https://www.diagrams.net/