BACKGROUND STUDY

To comprehend the domain of "HR - Employee Leave Management System," an exploration into its objectives and significance in organizational management was initiated. The study aimed to grasp the fundamental concepts of leave management and its role in fostering a balanced and efficient work environment. Additionally, a review of existing leave management systems and tools, such as those provided by industry leaders, shed light on best practices and functionalities.

PROJECT WORKFLOW

Based on the background study, the workflow of the Employee Leave Management System involves employees initiating leave requests through the system. Supervisors and HR managers receive notifications, review requests, and make decisions based on organizational policies. The system further incorporates features for calculating and updating leave balances and reporting tools for HR analytics.

REQUIREMENT ANALYSIS

The requirement analysis for the "HR - Employee Leave Management System" centers around identifying essential needs, functionalities, and objectives crucial for its effective implementation.

1. RoleTable:

Requirements:

Maintain a list of roles within the organization.

Each role is uniquely identified by a roleID.

Roles have names (roleName).

2. Employee Table:

Requirements:

Maintain information about employees.

Each employee is uniquely identified by an employeeID.

Employees have associated roles through the roleID foreign key.

Capture employee contact information such as email (email).

Store password information (password).

Track the creation time of employee records (create_time).

3. LeaveType Table:

Requirements:

Keep a record of different types of leaves available to employees.

Each leave type is uniquely identified by a leaveTypeID.

Each leave type has a name (leaveTypeName) and a description (leaveDescription).

4. LeaveRequestTable:

Requirements:

Track the leave history of employees.

Each leave request is uniquely identified by a leaveId.

Associate leave requests with specific employees using employeeID.

Associate leave request of a employee to a manager using managerID.

Record the start (startDate) and end (endDate) dates of the leave.

Link leave requests to a leave type through the leaveTypeID foreign key.

Capture the status (status) of each leave request.

5. LeaveComparisonTable:

Requirements:

Maintain a comparison of used and remaining leave durations for employees.

Each comparison entry is uniquely identified by a leaveComparisonID.

Associate comparisons with specific employees using employeeID.

Link comparisons to leave types through the leaveTypeID foreign key.

Record the used (usedDuration) and remaining (remainingDuration) leave durations.

These requirements provide a high-level overview of the information that the database is designed to capture and manage.

NORMAILIZATION

First Normal Form (1NF):

RoleTable:

The RoleTable has a primary key roleID, which is a simple attribute.

No repeating groups or arrays are present.

All columns hold atomic values.

Employee Table:

The Employee table has a primary key employeeID.

The roleID attribute appears to be atomic, and there are no repeating groups.

LeaveType Table:

The LeaveType table has a primary key leaveTypeID.

All columns hold atomic values.

LeaveRequestTable:

The LeaveRequestTable has a primary key leaveId.

The employeeID and leaveTypeID attributes appear to be atomic, and there are no repeating groups.

LeaveComparisonTable:

The LeaveComparisonTable has a primary key leaveComparisonID.

The employeeID and leaveTypeID attributes appear to be atomic, and there are no repeating groups.

PayrollDetails:

The PayrollDetails table has a primary key payrollID.The employeeID attribute appears to be atomic, and there are no repeating groups.

All tables seem to adhere to the first normal form (1NF) requirements.

Second Normal Form (2NF):

RoleTable:

There is only one candidate key (roleID). No partial dependencies.

Employee Table:

The Employee table has a primary key employeeID.

The roleID attribute is fully functionally dependent on the primary key.

LeaveType Table:

There is only one candidate key (leaveTypeID). No partial dependencies.

LeaveRequestTable:

The leaveTypeID attribute is fully functionally dependent on the primary key.

LeaveComparisonTable:

The leaveTypeID attribute is fully functionally dependent on the primary key.

PayrollDetails:

The employeeID attribute is fully functionally dependent on the primary key.

All tables adhere to the second normal form (2NF) requirements.

Third Normal Form (3NF):

RoleTable:

There is no transitive dependency. It's already in 3NF.

Employee Table:

The roleID attribute is directly dependent on the primary key.

LeaveType Table:

There is no transitive dependency. It's already in 3NF.

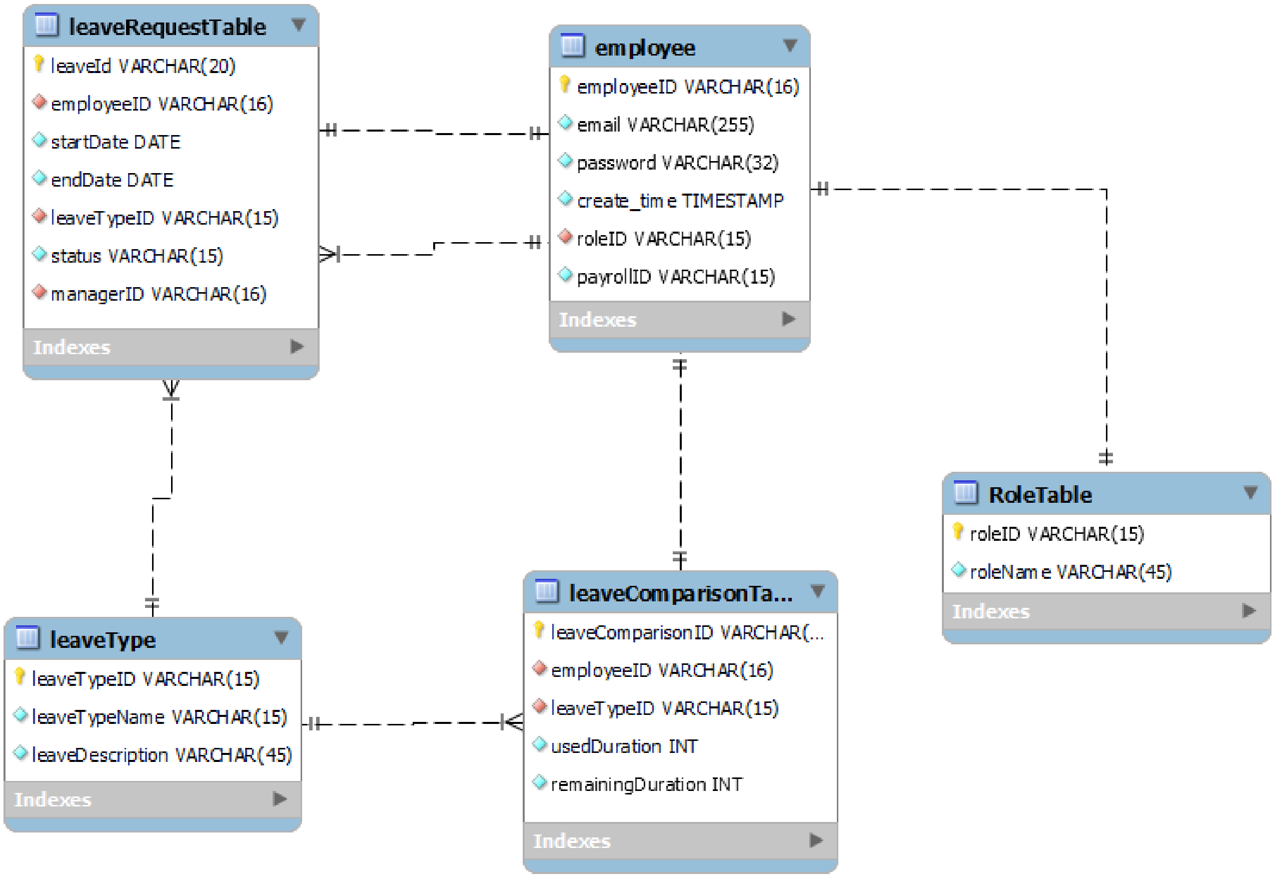
LeaveRequestTable:

There is no transitive dependency. It's already in 3NF.

LeaveComparisonTable:

There is no transitive dependency. It's already in 3NF.

ER DIAGRAM


leaveRequestTable
leaveId VARCHAR(20)
employeeID VARCHAR(16)
startDate DATE
endDate DATE
leaveTypeID VARCHAR(15)
status VARCHAR(15)
managerID VARCHAR(16)
Indexes
employee
employeeID VARCHAR(16)
email VARCHAR(255)
password VARCHAR(32)
create_time TIMESTAMP
roleID VARCHAR(15)
payrollID VARCHAR(15)
Indexes
RoleTable
roleID VARCHAR(15)
roleName VARCHAR(45)
Indexes
leaveType
leaveTypeID VARCHAR(15)
leaveTypeName VARCHAR(15)
leaveDescription VARCHAR(45)
Indexes
leaveComparisonTa...
leaveComparisonID VARCHAR(...
employeeID VARCHAR(16)
leaveTypeID VARCHAR(15)
usedDuration INT
remainingDuration INT
Indexes