**OPIM-5272-Business Process Modeling and Data Management-SECB14-1188**

**Team -6**

**Leave Management System**

Anisha Balakrishnan

Harish Babu Umapathy

Sivaraman Ramachandran

Hemanth Kumar Ponnambalam

***Introduction***

The aim of this project is to develop a leave management system that is of importance to any organization. The leave management system is an application that will handle leaves and store them using databases and Scan be accessed by every employee of the organization. It can be used to create automatic workflow of leave applications and its approvals. The automatic credit of leaves for every employee can also be achieved through this system. Auditing reports and various other reports can be retrieved using this system.

***Existing Process***

Paper based system where an employee registers a leave and gets it approved by his immediate supervisor. This form of process and management is time consuming and importantly prone to discrepancies. This also leads to loss of information and is an improper way to store leave information.

***Proposed Solution***

Due to the inefficiency and other related issues with manual leave system it is required to have an automatic computer-based application for process flow and store every detail using databases. This will streamline the leave process in an organization and help them store and handle leaves efficiently. It will help any organization perform the below key actions.

* Request for Leave
* Approve/Reject leave
* Maintains Employee leave record
* View leave balance
* View Pending requests/applications for employee and manager
* Update Leave balance based on job level
* View Leave Reports for a period
* Cancel leave application
* View Leaves per employee
* View Authorization record for each manager
* View Manager Dashboard/Employee Dashboard for company auditing

***About the Process:***

* *Triggering Event:* When an employee applies for a leave.
* *Major Interrelated Tasks include:*
* EMPLOYEE raising the leave request
* IT SYSTEM storing the request in the database
* MANAGER Approving/Rejecting the request
* HR MANAGER Approving/Rejecting the request
* IT SYSTEM updating the request status in the database
* AUDIT Team auditing total leaves approved/rejected/pending
* PAYROLL Team making salary deductions based on status on the Database
* *Specific Result*: Every Leave being tracked, and the entire process streamlined
* *Customer: - Organization where the Leave management system is incorporated.*

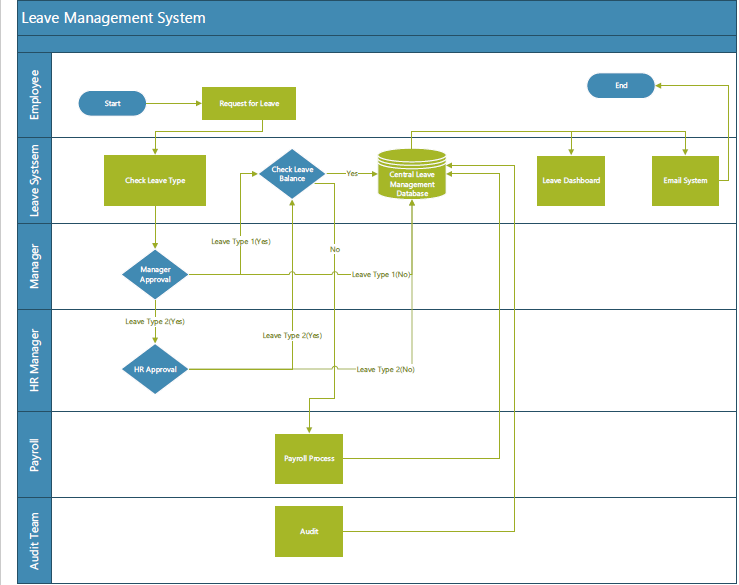
***Overview of the System***

Based on the type of leaves as below each leave request goes through multiple approvals and each step is tracked by the system and stored in the database for immediate or future viewing. The system first checks for leave balance for the employee and then forwards the request to the appropriate approver.



Note:- Leave level 1 will only need approval from immediate employee whereas type 2 will need second level from the employee’s HR manager and immediate manager. LOA leaves directly go to the payroll system for salary deduction.

***Swimlane Diagram***



There are in total 5 actors.

1. **Employee**: - Raises leave request and gets response via email. Leave request is stored in the central database
2. **Leave System: -** IT infrastructure where all applications and the main central leave management database is hosted.
3. **Manager**: - Also an employee who approves leaves for his subordinates in the Leave System and the details of which are stored in the database.
4. **HR manager**: - Based on the type of leave each leave request has multiple level approvals and the HR manager is the second level approval for Type 2 leaves. Once the leaves are approved by either manager/HR manager it gets updated in the database.
5. **Payroll Team:** - If it’s a loss of pay type leave the payroll team will make the employee eligible for salary deductions. This information is also stored in the database
6. **Audit Team:** - Will be able to view reports for each manager and employee to see for discrepancies and correctness (valid approval/valid request etc.) for each leave request. Report will be taken from the information stored in the database.

**Note**: - This project is only limited to storing and retrieving information from the database. Actual updates to the database are to be done by other IT systems/applications.

***Issues without the Process:***

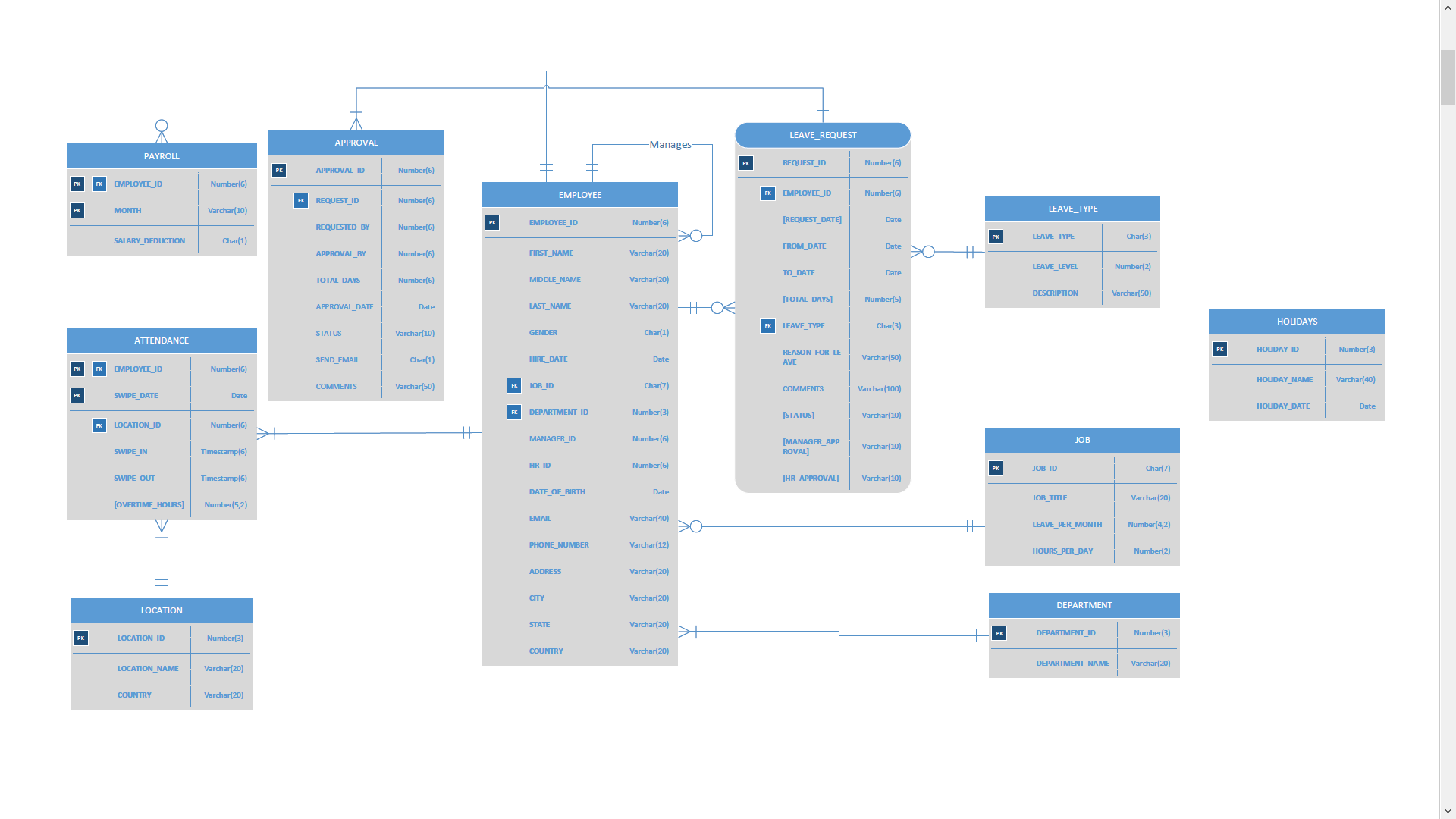
* All the information related to the employee and their leave requests is currently being maintained in manually which is an inefficient way of data storage and data retrieval.
* Daunting task for managers to view and approve requests
* Without centralized and interlinked processes, the leave system is prone to have manual errors and chances for misusage of leaves are high
* No control over number of leaves taken and approved for employee and manager respectively.

***Solutions Implemented & Benefits Obtained:***

* Every Leave request is stored and readily retrievable at any point in time
* No data duplication. This has been achieved by giving relevant primary key constraints.
* Data integrity achieved by implementing relevant constraints to each attribute in the tables.
* Overall reduction in process time
* Views created to ease data retrieval
* Triggers created to automatically make changes to the number of leaves taken/available etc. for each employee
* Each employee/manager/audit Team have greater visibility to the entire process as at each stage a leave request can be tracked, and the information can be viewed.

***DATABASE DESIGN:***

***Entity Relationship Diagram***



ERD diagram



***Entity Description based on Business Rules:***

The entities and attributes of the database are as follows:

*Employee*: - An entity employee is created to store information of every employee details which includes personal profile, address, hire date, job id, department id, manager id, HR id. We can retrieve all the information related to employee from this entity

*Leave Request*: - An entity leave request is created to store information of employee’s leave request details. Some of the important Attributes includes request ID, Emp ID, leave duration date, type of leave employee applied, reason for leave. Each employee’s leave request will be stored in this entity

*Leave Type*: - An entity leave type is created to store information about leave types, leave level and description at an organizational level. Leave types are Medical Leave (Level 1), Bereavement Leave (Level 1), Leave of Absence (Level 2), Maternity Leave (Level 2), Paternity Leave (Level 2), Earned Leave (Level 1) and Sabbatical Leave (Level 2).Level 1 signifies that only Manager approval is required whereas Level 2 implies 2-level approval which is manager and HR approval.

*Job:* - An entity Job is created to store information about Job id, title of the job, work hours for that related job and total leaves per month applicable for that job level. Job title includes Trainee, System Engineer, Technical Lead, Manager, Senior Manager.

*Department:* - An entity department is created to store various departments in the organization and they are Finance and Accounts, HR, Admin, IT, Marketing, R&D.

*Holidays: -* An entity Holidays is created to store information about holidays, name of holidays and its date. Some of the holidays includes New year, Independence Day, Labor Day, Thanks giving, Christmas and other few holidays.

*Approval:* - An entity Approval is created to store information about approval ID, request id, manager’s approval name, date of approval and status of approval. In this entity approval id is created for every employee’s leave request. Authority of Approval could either be manager or both HR and manager, which depends on type of leave employee applied.

*Payroll:* - An entity payroll is created to store information about employee Id, month and salary deduction. Salary would be deducted if the employee takes loss of leave type(SBA-Sabbatical Leave/LOA-Leave of Absence).

*Attendance:* - An entity attendance is created to store information about employee attendance details. Attributes includes Emp Id, Swipe date, Location ID, Swipe in and swipe out. For Overtime hours indicates that an employee has worked more than required hours for that job level.

*Location:* - An entity location is created to store information about location details. Location attributes are location Id, location name and country. Location name includes Florida, New York, Texas, California, Georgia and Colorado. These are the locations that an employee might work.

***Relationships:***

* Any employee in the organization can take leave by requesting for leave. Every employee working in the organization is managed by a Manager and has an HR Manager. An employee can either take no leave or any number of leaves by raising a request for leave. A leave request can be raised by only one employee at a time. Employee can also be supervisor for many employees but not all employees are supervisors.
* Every leave request is checked for its Leave type. Every leave type can be categorized into two Leave levels – Level 1 leave and Level 2 leave. Leave Level 1 requires the approval of the employee’s Manager and Leave Level 2 requires the approval of the employee’s Manager and HR Manager as well.
* Holiday is a standalone entity as it has no relationship with other entities. It is used to calculate the total number of business days the employee requested for leave based on from date and to date of leave request. There are 10 Public holidays per year which are common to all the employees in the organization irrespective of their job title.
* A leave request can be any of the leave type available of any level but only one leave at a time.
* One leave request will have either one or two approvals. Leaves under Leave Level 1 category require one approval from the employee’s Manager. Leaves falling under Level 2 category require two approvals, one from the employee’s Manager and the HR Manager.
* An employee who is present for work will have attendance on a daily basis. Thus, every employee will have one or more days of attendance. Weekends are not considered as a part of working days. On the basis of an employee’s job title, the number of overtime hours per day is tracked along with the attendance.
* Payroll is generated for employee with salary deduction flag set to Y where the leave type belongs to SBA/LOA. One employee can have zero or more pay slips and not all employees
* Every employee has a job title assigned to him or her. More than one employee can have the same job title but one job title per employee at a time. Depending on the job title, every employee will have a fixed number of working hours per day and leaves per month.
* A department can have multiple employees working in it. But an employee can belong to only one Department at one instance.
* A location can have more than one employee working in it. Thus, attendance of every employee working in every location is tracked. One employee can swipe in and swipe out on a day in only one location.

***Database Entities Relationship:***

Excel sheet provides the list of all DB entities and their descriptions, constraints, process rules.



***Automate Process for Leave Management System:***

Triggers have been implemented wherever feasible to automate the process associated with the leave management system

1. ATTENDANCE\_TRG: - Calculates Overtime hours based on swipe in, swipe out details of the employees
2. REQUEST\_PK\_TRG: - Generates Sequence number for primary key request id
3. REQUEST\_TRG: - Checks whether the employee has leave balance when raising the leave request and triggers the one level approval/two level approval.
4. REQUEST\_TRG1: -Creates entry in the approval table for every entry in the leave\_request table based on manager and HR approval status.
5. APR\_SEQ\_TRG: - Generates Sequence number for primary key approval id
6. APR\_TRG: - Checks if first level approval is done and throws error if it goes to next level without first level approval
7. APPROVAL\_TRG: -Updates the leave\_request table with the status based on the action taken by the manager. For example, below query when run by user/application upon approval/rejection will trigger the change in leave request status through this trigger

*UPDATE APPROVAL SET STATUS='Approved',Comments='Approved the Leave' where request\_id=&request\_id and approval\_by = &approval\_by;*

***Queries to obtain important business reports:***

The use of databases facilitates the creation of reports which are useful to define improved business processes. A few reports and their usage have been described below

1. Report to provide the details of all mangers in the organization and their employees list.

This report is useful for the organization to get the list of all employee names who is the manger to one or many employees in the organization. Also, this report lists all the employee details of the manger in a single row for each manger and also the number of employees supervised by that manager.

*select e1.first\_name||' '||e1.last\_name "Manager", LISTAGG(e2.first\_name||' '||e2.last\_name, ',')*

*within group (order by e2.first\_name||' '||e2.last\_name) "Employees",Count(\*) "EmployeesCount" from employee e1 join employee e2*

*on e1.employee\_id = e2.manager\_id*

*group by e1.first\_name||' '||e1.last\_name;*

1. Report to provide the number of leave requests per leave type raised each year

This report will be useful for the audit team to track the number of leaves taken per leave type every year and their trend across years and this can be helpful in modifying leave rules and leave crediting. Leave Level is a user entered parameter which can be 1/2.

*select TO\_CHAR(REQUEST\_DATE,'YYYY') "RequestYear",leave\_type "LeaveType" ,count(request\_id) "Numberofrequests" from*

*leave\_request where leave\_type in ( select leave\_type from leave\_type where leave\_level = &a)*

*group by leave\_type,TO\_CHAR(REQUEST\_DATE,'YYYY');*

1. Report to provide the dashboard details for the manager

This report will list the total number of leaves approved/rejected/pending for each manager and also the average ageing days of the pending requests. This query will be the Manager dashboard and can be used by the manger to track his details over a certain period of time. This can also be used by the audit team to check if the approvals are done and take necessary action if there are any pending approvals.

*select apr.approval\_by "Manager ID",emp.FIRST\_NAME "Manager Name",*

*sum(case*

*when apr.status = 'Pending' THEN 1*

*else 0*

*end ) "Pending", AVG(trunc(sysdate-request\_Date)) "AveragePendingAgeingDays",*

*sum(case*

*when apr.status = 'Approved' THEN 1*

*else 0*

*end ) "Approved",*

*sum(case*

*when apr.status = 'Rejected' THEN 1*

*else 0*

*end ) "Rejected",Count(\*) "Total Requests"*

*from approval apr,employee emp,leave\_request req where apr.approval\_by=emp.employee\_id and req.request\_id=apr.request\_id*

*and req.REQUEST\_DATE between TO\_DATE(&f,'MMDDYYYY') and TO\_DATE(&t,'MMDDYYYY')*

*group by apr.approval\_by,emp.FIRST\_NAME;*

1. Leave Balance View for the Employees

This report is used by the employees to check for the number of leaves they earned and accrued. Also, the number of overtime hours and their current leave balance.

*CREATE VIEW LEAVE\_BALANCE*

*AS*

*SELECT Employee\_id,TotalEarnedLeave,TotalEarnedLeaveYTD,TotalAccruedLeave,TotalAccruedleaveYTD,OVERTIME\_HOURS,*

*TotalEarnedLeave-TotalAccruedleave+OVERTIME\_HOURS AS LEAVE\_BALANCE FROM (*

*SELECT Employee\_id,NVL((select sum(total\_days)*

*from leave\_request where status = 'Approved' and employee\_id=A.employee\_id*

*group by employee\_id),0) AS TotalAccruedLeave,NVL((select sum(total\_days)*

*from leave\_request where status = 'Approved' and employee\_id=A.employee\_id and request\_date >= TRUNC(SYSDATE,'YY')*

*group by employee\_id),0) AS TotalAccruedleaveYTD,*

*trunc(months\_between(sysdate,A.hire\_date) \* B.leave\_per\_month) AS TotalEarnedLeave,*

*trunc(months\_between(sysdate, TRUNC(SYSDATE,'YY')) \* B.leave\_per\_month) as TotalEarnedLeaveYTD*

*,NVL((SELECT SUM(OVERTIME\_HOURS)/8 FROM ATTENDANCE WHERE EMPLOYEE\_ID=A.EMPLOYEE\_ID),0) AS OVERTIME\_HOURS*

*FROM EMPLOYEE A,JOB B WHERE A.JOB\_ID=B.JOB\_ID);*

1. Report to analyze the employees overtime hours across all locations.

This report will provide the average overtime hours for each location of the organization and this will be helpful for the audit team to analyze the employee’s Over Utilization across locations.

*SELECT B.LOCATION\_NAME,TRUNC(AVG(a.overtime\_hours),2) "AverageOvertimeHours" FROM ATTENDANCE A,LOCATION B WHERE A.LOCATION\_ID=B.LOCATION\_ID group by B.LOCATION\_NAME;*

***Database Implementation Steps:***

* Below script needs to be executed for building the database sequence, table and views.



* Below script needs to be executed to load sample data for testing purpose



* Below script needs to be executed implement triggers in the database that automates and simplifies the process.

