Human Resource Management System – Requirements Document

# Day 3 – Java Setup

Objective: Implement Java-based Employee CRUD operations, employee authentication, and full CRUD for Payroll, Leave Requests, Departments, and Attendance using DAO classes for direct database access.

# Tasks Completed

1. Employee CRUD Implementation (EmployeeDAO)
   * Add Employee (

**addEmployee(Employee emp)**

password, returns auto-generated ID.

) – Inserts a new employee record, stores

* + View All Employees ( **getAllEmployees()** ) – Retrieves all employees with department names.
  + Get Employee by ID ( **getEmployeeById(int id)** ) – Fetches a single employee with department name.

**updateEmployee(Employee emp)**

* + Update Employee ( password.

) – Updates employee details including

* + Delete Employee ( **deleteEmployee(int id)** ) – Deletes employee by ID.
  + Search Employee by Name ( **searchEmployeeByName(String name)** ) – Case-insensitive search.
  + Get Employees by Department ( **getEmployeesByDepartment(int departmentId)** ).

1. Employee Authentication
   * Login Validation ( **validateLogin(int empId, String password)** ) – Checks credentials, returns **Employee** object if valid.
2. Payroll Management (PayrollDAO)
   * Add Payroll ( **addPayroll(Payroll p)** ) – Inserts payroll data for a specific employee.
   * Get All Payrolls ( **getAllPayrolls()** ) – Retrieves all payroll records, sorted by pay date.
   * Get Payroll by Employee ( **getPayrollByEmployee(int empId)** ) – Fetches payrolls for a specific employee.

**updatePayroll(Payroll p)**

* + Update Payroll ( net salary.

) – Updates salary components, deductions, and

* + Delete Payroll ( **deletePayroll(int id)** ) – Deletes a payroll record by ID.
  + Utility Method ( **mapResultSetToPayroll(ResultSet rs)** ).

1. Leave Request Management (LeaveRequestDAO)
   * Add Leave Request (

**addLeaveRequest(LeaveRequest l)**

with start/end dates, type, and status.

**getAllLeaveRequests()**

* + View Leave Requests (

) – Stores employee leave request

,

**getLeaveRequestsByEmployee(int**

) – Fetch all leave requests or by employee ID.

**empId)**

**updateLeaveRequest(LeaveRequest l)**

* + Update Leave Request ( Rejected) or dates.

**deleteLeaveRequest(int id)**

* + Delete Leave Request (

) – Update status (Approved/

) – Remove leave request by ID.

1. Department Management (DepartmentDAO)
   * Add Department ( **addDepartment(Department d)** ) – Insert a new department.
   * View All Departments ( **getAllDepartments()** ) – Retrieve all department records.
   * Get Department by ID ( **getDepartmentById(int id)** ) – Fetch department details by ID.
   * Update Department ( **updateDepartment(Department d)** ) – Update department name/ location.

**deleteDepartment(int id)**

* + Delete Department ( ) – Remove a department by ID.

1. Attendance Management (AttendanceDAO)
   * Mark Attendance ( **addAttendance(Attendance a)** ) – Record daily attendance for an employee.
   * View Attendance ( **getAllAttendance()** , **getAttendanceByEmployee(int empId)** ) – Retrieve attendance records, optionally by employee.
   * Update Attendance ( **updateAttendance(Attendance a)** ) – Modify attendance status.
   * Delete Attendance ( **deleteAttendance(int id)** ) – Remove attendance record by ID.
2. Main Application Integration
   * Console menu updated for Payroll, Leave Requests, Departments, and Attendance.
   * CRUD options implemented for each module.
   * Payroll menu includes generate, view, update, delete payrolls.
   * Ensures proper interaction with DAO methods and calculates net salary dynamically.
3. Database Integration
   * All DAOs use for SQL database connectivity.

**DBConnection.getConnection()**

* + Prepared statements used to prevent SQL injection.
  + Try-catch blocks handle SQL exceptions and log errors to console.

1. Next Steps
   * Complete integration of all services (PayrollService, LeaveService, AttendanceService).