Payroll System Development Plan

1. Overview

The Payroll System is designed to manage employee records, including salary details, attendance, and roles. It will be built using **Java Swing/AWT** for the frontend and **MySQL** for the backend database.

2. User Roles & Login

- **Admin Login**: Access to all functionalities including employee management and payroll modifications.
- **Employee Login**: Limited access to view salary details, attendance, and request updates (if required).

3. Core Features

3.1 Employee Management

Create New Employee Record

- Employee ID (Auto-generated)
- Name
- Age
- Date of Birth (DOB)
- Date of Joining (Default to Current Date)
- Salary
- Role
- Bank Account Details (Bank Number)

Delete Employee Record

- Employee record deletion by ID.

Update Employee Record

- Modify Salary
- Update Role

Display Employee Record (Read Operation)

- View **Basic Details** of all employees.
- View **Attendance** records of employees.

4. Database Design

Tables and Fields

1. Employee Table

Column Name	**Data Type**	**Description**
employee_id	INT (Primary Key, Auto Increment)	Unique ID for each employee
name	VARCHAR(100)	Employee's full name
age	INT	Employee's age
dob	DATE	Date of Birth
date_of_joining	DATE	Default to current date
salary	DECIMAL(10,2)	Salary of the employee
role	VARCHAR(50)	Employee's role
bank_no	VARCHAR(20)	Bank account number

2. Attendance Table

Column Name	Data Type	Description
attendance_id	INT (Primary Key, Auto Increment)	Unique ID for attendance record
employee_id	INT (Foreign Key)	References Employee Table
date	DATE	Date of attendance
status	VARCHAR(10)	Present/Absent

5. Tech Stack

- **Frontend:** Java Swing/AWT
- **Backend:** MySQL Database
- **IDE:** IntelliJ IDEA / Eclipse
- **JDBC Driver:** MySQL Connector

6. Implementation Plan

Phase 1: Database Setup

- Design and create tables in MySQL.
- Establish database connection in Java using JDBC.

Phase 2: GUI Development

- Create a user-friendly interface using Swing/AWT.
- Implement input forms for adding and updating employee details.

Phase 3: CRUD Operations

- Implement **Create**, **Read**, **Update**, and **Delete** functionalities for employee records.
- Integrate salary and role update features.

Phase 4: Attendance Management

- Implement attendance tracking system.
- Display attendance records in the admin panel.

Phase 5: Testing & Debugging

- Perform unit testing on database queries.
- Validate GUI interactions and functionality.

7. Future Enhancements (Optional)

- Payroll calculation and tax deductions.
- Monthly reports generation.
- Email notifications for salary slips.

Team Notes:

- Ensure proper exception handling for database connectivity issues.
- Use MVC (Model-View-Controller) pattern for structured code.
- Follow best practices for UI design to ensure a smooth user experience.

This document should serve as a **guide** for your team while developing the Payroll System.