

Human Resource Management System

Your team is planning to start a software product based startup. The initiative was taken by requirement of a major apparel company, Jupiter. Jupiter Apparels is a multinational corporation with over 1000 employees. Currently the branches of Jupiter are at Sri Lanka, Bangladesh and Pakistan. Although they use SAP as the ERP to manage key activities of the organization, the management has felt that SAP is too much of an overhead in a management perspective. On one hand it requires specifically trained SAP engineers to do the slightest change while training management employee to use the system is a hassle. Hence, Jupiter has requested your team come up with a easy to use human resource management system as the initial phase. If it succeed, Jupiter is planning to move all other tasks including payroll, inventory, etc. to your system.

Your team has seen the opportunity to create a HRM product that could be marketed across several companies in future. In order to get an idea of the HR processes, your team went to head office of Jupiter in Union place to observe the activities. The system in its initial phase is required to support PIM (personal information management) and Absence management. The product is meant to be supporting one specific organization. Settings such as organization name, address, registration number etc. are stored there. The goal of the PIM module is primarily to store information about the employees. Extensive information regarding employee including name, birth date, marital status etc. are included in PIM. Another vital information is the dependent information of the employee which is required for payroll functions. (not currently part of the system). Emergency contact details are also need to be store per employee. In the organizational structure perspective, an employee has job title (HR Manager, Accountant, Software Engineer, QA Engineer), a pay grade (Level1, Level2, ..) and an employment status (Intern (fulltime, parttime), Contract (fulltime, parttime), Permanent, Freelance). These must be defined independently from the employee so that more fields can be added / removed in future. Additionally each employee has one supervisor, to whom he/she reports to and can have multiple subordinates. While these are the very basics of any organization's employee information, your team soon understood that the **product needs to have the capability to define new custom employee attributes** in future. For an example, the field "Nationality" can be important to some organization, but the product should not explicitly define it, but rather provide the user with the freedom to do so. The system always has one admin user. The user account is used to create the second management user, HR manager. Afterwards, HR manager can continue adding rest of the employees to the system. Each user of the system must be bind to an employee record. But it is possible to have employee records without a user.

Apart from PIM, the product needs to implement an Absence module which provides leave related functionality. Jupiter's leave structure is static with 4 types of leaves; **Annual**, **Casual**, **Maternity** and **No-pay**. **Depending on the pay grade**, the number of leaves of each category varies. But every employee has **mandatory 50 no-pay leaves**. These properties must be configurable. When an employee need to add a leave, he/she should login to the system and complete the leave application form. Once submitted, the request for leave is sent to the

supervisor of the employee. The supervisor must login to the platform and approve the leave to process the leave. Once approved only, the leave would be deducted from the employee's leave counts. A comprehensive set of reports are also required by Jupiter. And your team plans to modularize it to a separate reporting module. At minimum the following reports are requested,

- Employee by department
- Total leaves in given period by department
- Employee reports grouped by job title, department, pay grade etc.
- Reports based on custom fields (should contain at least 2 custom fields created beforehand)

Finally in the user management perspective of the system, fine grained authorization is required for every module and sub-module of the system. For an example, the system would limit the access of a level 1 employee to only view his personal information and nothing else. He/she will not be able to edit own information either. A managerial employee can be given edit access to all PIM information but no access to Absence related functionalities.

In order to initiate the product, your team has decided to first produce a well thought out database design. And then implement a PoC program to test the functionalities of the database. At this stage the UI/UX aspects are irrelevant.

Task

Your task is to model the database design to encapsulate these requirement. It should consider all entities and relationships given in the description. Moreover you need to identify the places where **procedures, functions and triggers** can be employed to guarantee ACID properties. Foreign keys and primary keys must be set to maintain consistency. Indexing should be done when necessary.

Additionally, the you must get a domain idea by reading related material and take assumptions when not explicitly provided. The database **must be** populated with some employee data and users beforehand. Pick a sample organization and enter the data (organization information, organization structure, leave types, job titles etc.) into the database. These data insertions can be done manually and no need of UI components just for the task of data input.