STUDENT : Popa Ioan-Ciprian

PROJECT for SOFTWARE ENGINEERING LABORATORY

APPLICATION TITLE Employee Management System

1. APPLICATION DESCRIPTION

This project is about building a basic employee management system easy to use and accessible by

multiple roles of employees.

Employees can use the system for:

- Referring the post to unemployed people

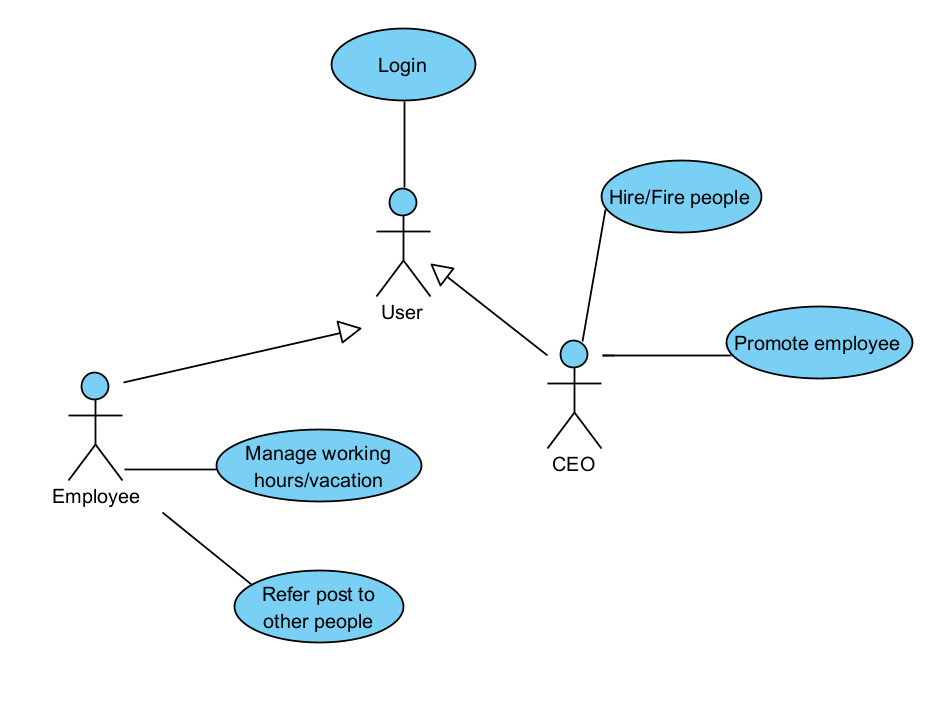
- Manage their working hours/vacation

CEO’s can use the system for:

- Promoting/demoting people’s post to a higher/lower rank

- Hire/Fire employees

2. USE CASE DIAGRAM



3. USE CASE DETAILS : (For each UC)

UC name: Refer Employee

Actor(s) : Employee

Description : This use case describes the process by which an employee can refer another person for a job within the company.

Preconditions : - The employee must be logged into the system.

* The employee must have the necessary permissions to refer other individuals for job positions.
* The employee must know the details of the person they want to refer, including their contact information and relevant qualifications.

(Sequence diagrams at system level, specifying which flow is modelled in each of them, main

flow and/or alter native flows)

1.The employee logs into the system.

2.The employee navigates to the "Refer Employee" section of the system.

3.The system presents a form for the employee to input the details of the person they want to refer, including name, contact information, and qualifications.

4.The employee fills out the form with the required information.

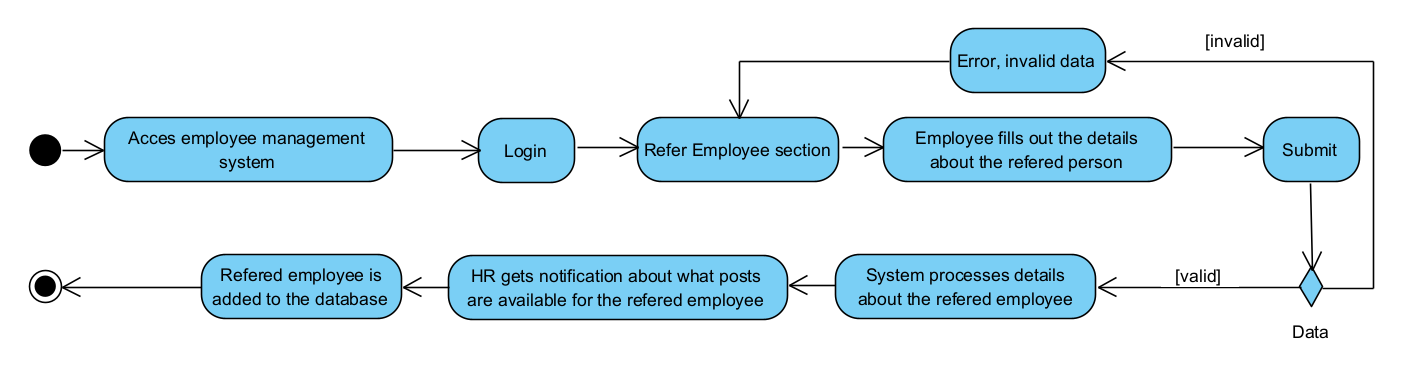
5.The employee submits the referral.

6.The system processes the referral and notifies the HR responsible for recruitment.

7.The referred individual's details are added to the recruitment database for further consideration.

Postconditions : - The referral is recorded in the system.

* The referred individual's details are available for review by the recruitment team.
* The employee receives a confirmation message indicating that the referral has been successfully submitted.

Alternative flows: - If the employee enters invalid information in the referral form (e.g., missing required fields, incorrect contact information), the system displays an error message.  


Use Case Name: Manage Working Hours/Vacation

Actor(s): Employee

Description: This use case describes the process by which an employee can manage their working hours and request vacation time within the company.

Preconditions: - The employee must be logged into the system.

* The employee must have the necessary permissions to manage their working hours and request vacation time.
* The employee's current working schedule and available vacation balance should be up-to-date in the system.

Sequence Diagram (Main Flow):

1.The employee logs into the system.

2.The employee navigates to the "Manage Working Hours" or "Request Vacation" section of the system.

3.If managing working hours:

- The system displays the employee's current working schedule.

- The employee selects the option to modify their working hours.

- The employee adjusts their working hours by specifying new start and end times or adding/removing shifts.

- The employee submits the changes.

- The system updates the employee's working schedule accordingly.

4.If requesting vacation:

- The system displays the employee's available vacation balance.

- The employee selects the option to request vacation time.

- The employee specifies the dates for their vacation request.

- The employee submits the vacation request.

- The system processes the request and checks for conflicts with existing schedules or company policies.

- If the request is approved, the employee's vacation balance is adjusted, and their absence is recorded in the system.

- If the request is denied, the employee receives a notification with the reason for denial.

Postconditions: -For managing working hours:

* The employee's working schedule is updated according to the changes made.

For requesting vacation:

* If approved, the employee's vacation balance is adjusted, and their absence is recorded.
* If denied, the employee's vacation balance remains unchanged, and they receive a notification of denial.

Sequence Diagram (Alternative Flow - Conflicting Vacation Request):

1.If the requested vacation dates conflict with existing schedules or company policies:

- The system notifies the employee of the conflict.

- The employee revises the vacation request with alternative dates.

- The system reevaluates the request based on the new dates.

- The process continues until a non-conflicting request is submitted.

4. ACTIVITY DIAGRAMS : (For 2 use cases)

UC name:................

The activity diagram

UC name:................

Activity diagram

5. GUI PROTOTYPE

Starting with the initial screen of the application, represent an initial screen for each use case

that is connected to an actor, screens content and navigation flow (using State Machine

Diagram).

If use cases with more screens exist, for each of such use case (after you write the name of

the use case) represent the screens, their content and the navigation flow (using State Machine

Diagram).

UC name:................

Screens, their content and the navigation flow (represented using State Machine Diagram).

6. DOMAIN MODEL

Class diagram containing domein model : classes and relations (possibly some attributes).

7. ROBUSTNESS DIAGRAMS

The robustness diagrams for 2 use cases. It must be choosen at least one complex use case,

that implies more than simple data transfers between the system (application) and the user.

UC name:................

Robustness diagram

UC name:................

Robustness diagram

8. SEQUENCE DIAGRAMS

The sequence diagrams for the same 2 use cases.

UC name:................

Sequence diagram

UC name:................

Sequence diagram

9. THE EXTENDED CLASS DIAGRAM

The class diagram resulted from the robustness analysis of the 2 use cases.