Vacation Tracking System - VTS

DOCUMENTATION

ABDALLA KHALIFA

Vision

The main goal of this system is to improve the internal process of the organization respecting the time it takes to manage vacation time, providing individual employees, the ability to manage their own vacation time, sick leave and personal time off without knowledge in company policy.

Actors & Activities

We have four actors in the scene:

1- **Employee**: Manage Time [Create new requests].

2- **Manager**:

   i.   Approve requests.

   ii.   Award time.

3- **HR**:

   i.   Edit employee record.

   ii.   Manage locations.

   iii.   Override leave records.

   iv.   Manage leave categories [sick leave etc..].

4- **System Admin**: Backup system logs.

Functional Requirements

- Rule based system.

- Enables manager to approve vacation requests [optional].

- E-mail notification [Manager – Employee].

- System should use the portal's single – sign – on mechanisms for all authentications.

- Activity logs for all transactions [Auditing].

- HR and system admin have the ability to override rules & Auditing.

- Enables manager to directly award personal leave time according to system roles.

- Integration services.

Non – Functional Requirements

- **Usability**: The system must be easy to use giving the user a smooth and simple experience.

- **Scalability**: The system should have the ability to scale and add new features and integrate with systems related to the facility.

Constraints

- The system uses the company existing hardware and middleware [legacy hardware].

- The system is extension to the existing intranet portal system.

- The system uses the portal's single-sign-on mechanism for all authentications.

- All system transactions are saved in the system logs.

- User experience.

Business Roles

- All employees work eight hours a day.

- Each employee's vacation time requests are subject to the restrictions of each employee's primary work location in addition to overall company policies and restrictions.

- Vacation requests validation rules are defined and owned by the HR department.

- An employee can't take more than x consecutive days of leave for Y type of grant.

- Vacation time of type X cannot be taken when directly adjacent to a company or location – specific holiday.

- Vacation time may not be granted when there is only x number of employees scheduled to work from a predefined list of Y employees (there is short of employees count).

- Vacation time may not be granted on specific dates.

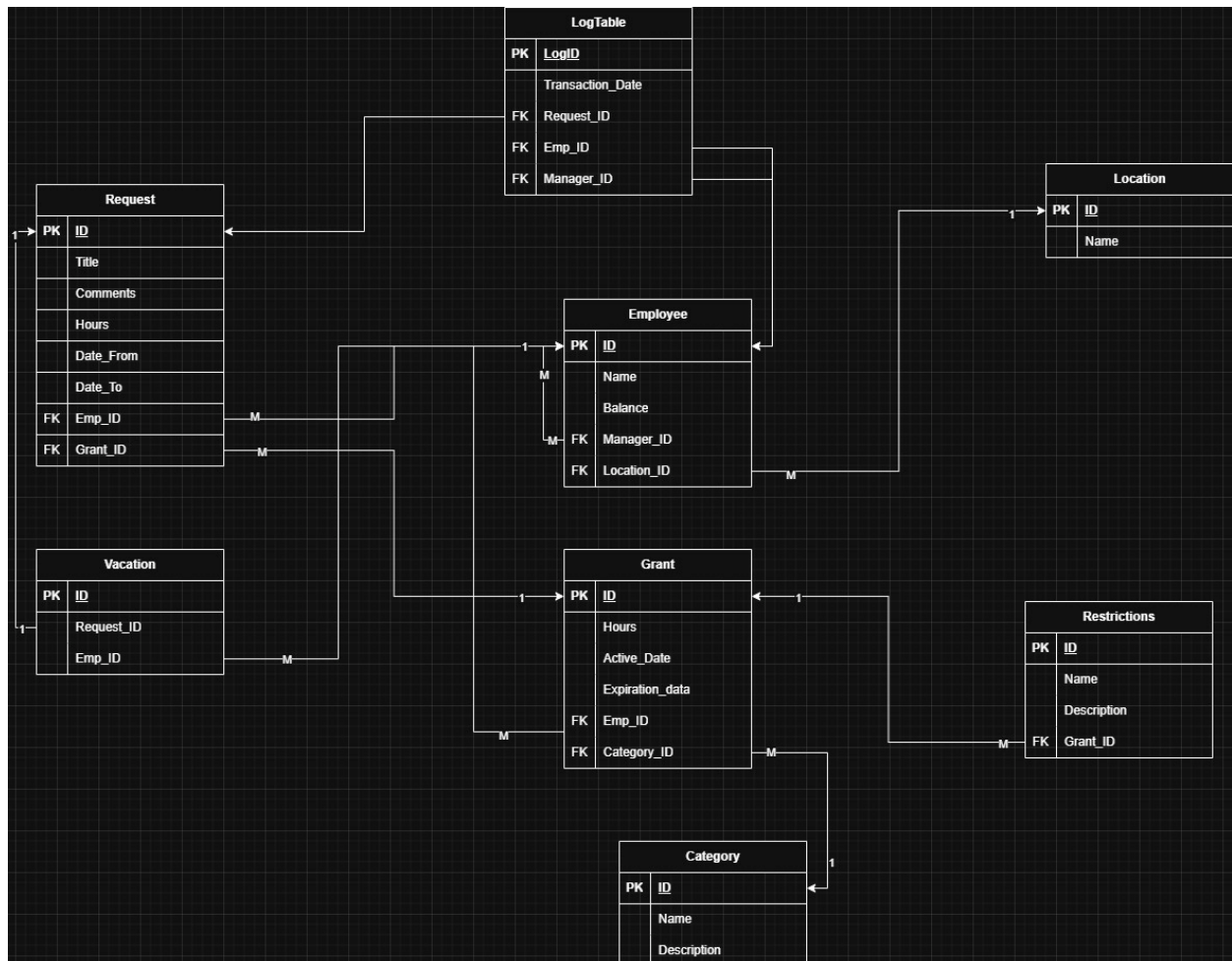- Vacation time of a certain type is limited to certain days of the week.

Note: All specific constraints are in the company policy maintained by the HR department.
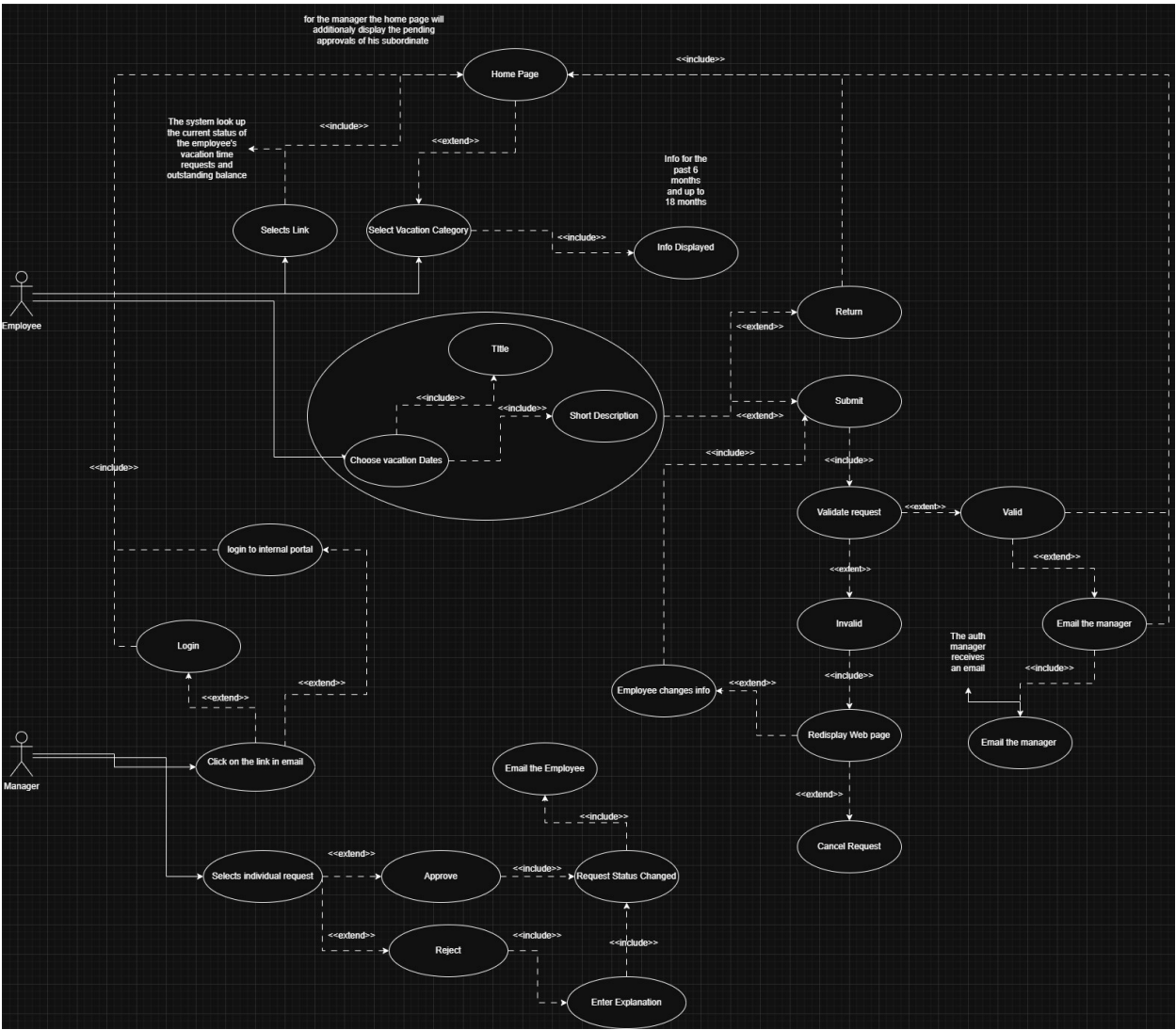
Assumptions

- The company's existing software is scalable and have the ability to integrate with new made extensions.

- The company's infrastructure is sufficient and supports our needs.
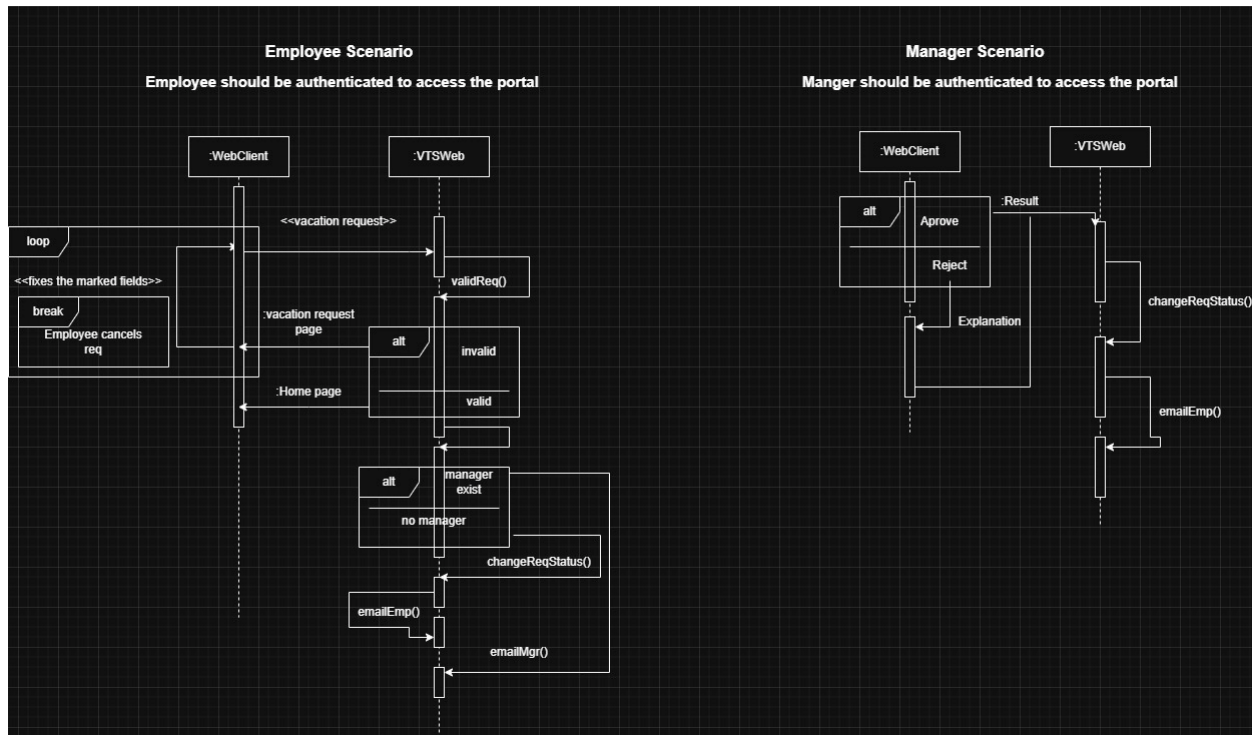
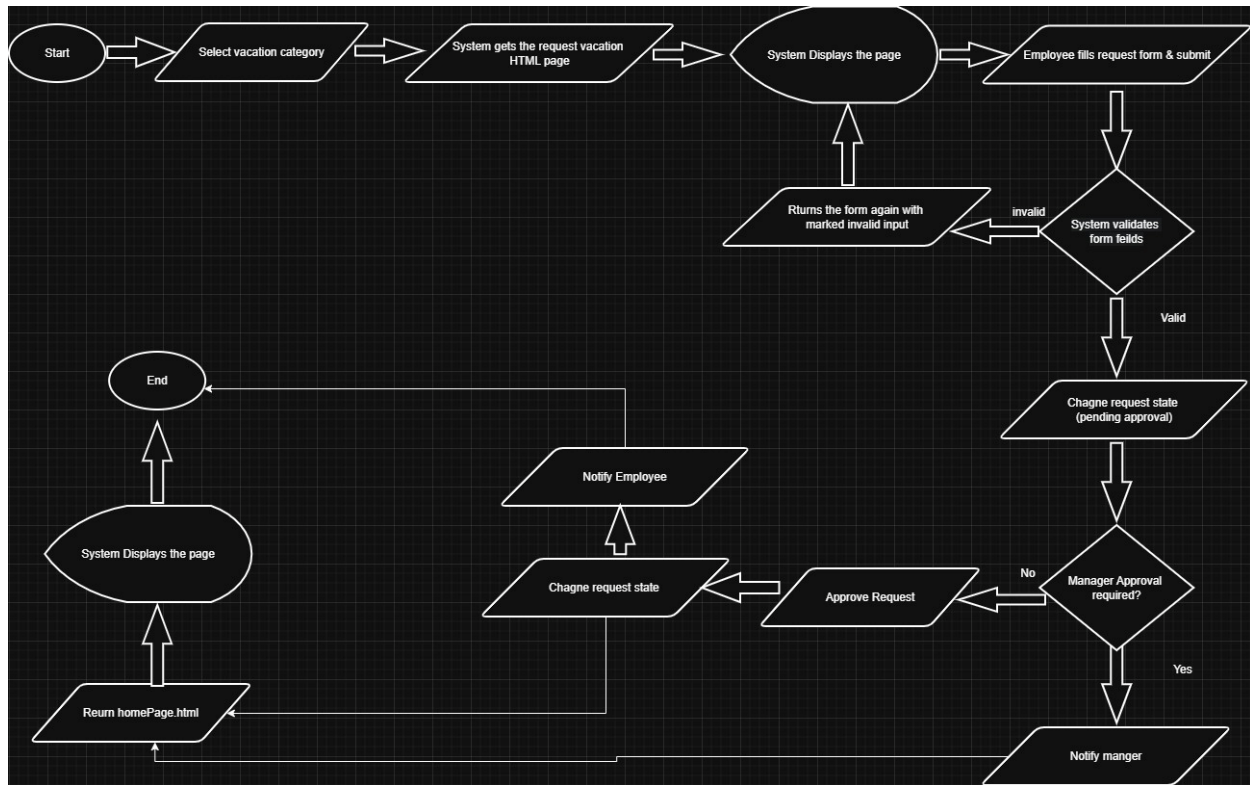---

Data Model

Use Case Diagram

Sequence Diagram

Employee Scenario

Employee should be authenticated to access the portal

:WebClient   :VTSWeb

<<vacation request>>

loop

<<fixes the marked fields>>

break

Employee cancels req

validReq()

:vacation request page

alt

invalid

valid

:Home page

alt

manager exist

no manager

changeReqStatus()

emailEmp()

emailMgr()

Manager Scenario

Manger should be authenticated to access the portal

:WebClient   :VTSWeb

alt

:Result

Aprove

Reject

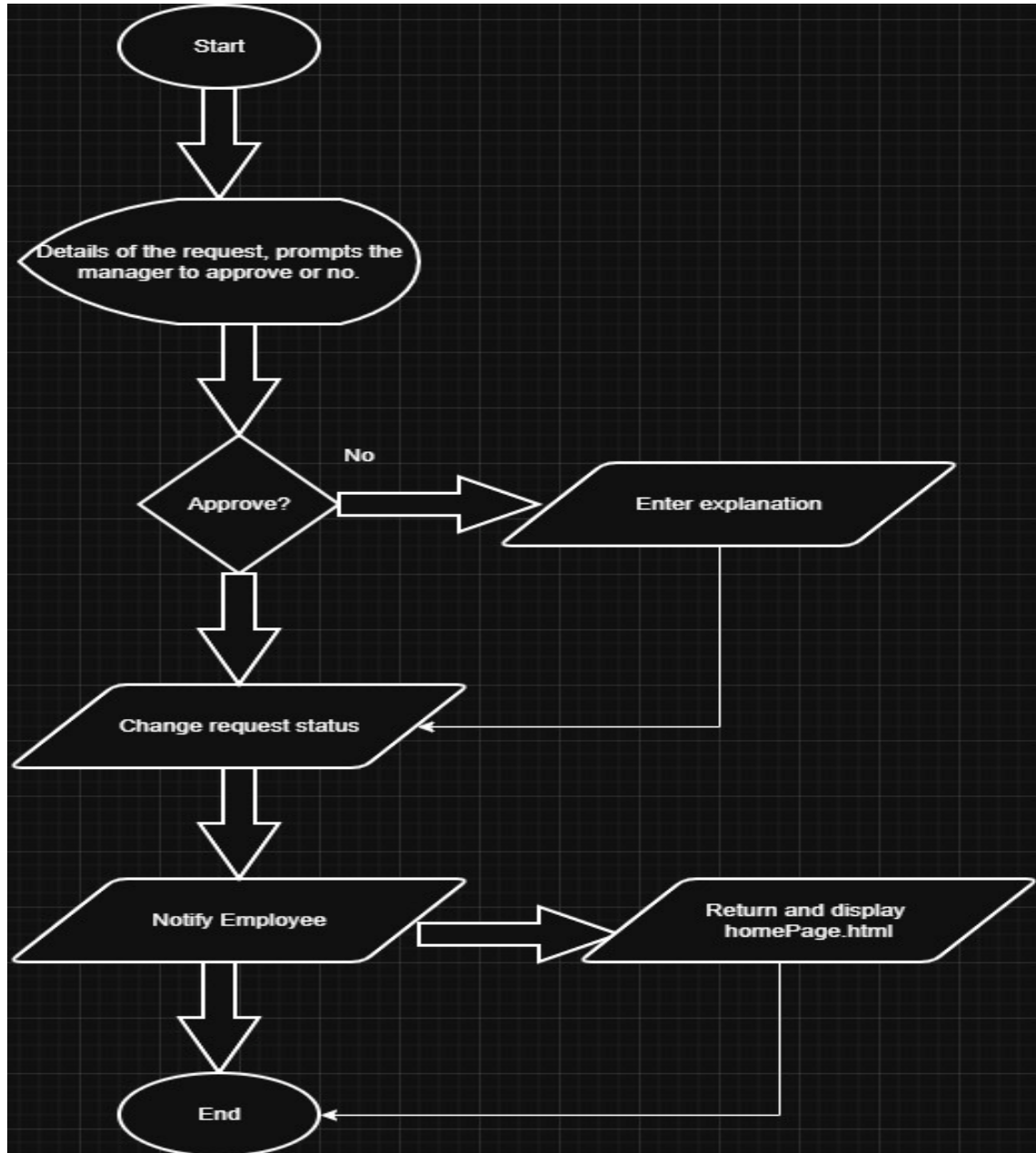changeReqStatus()

Explanation

emailEmp()

Create New Request (Flowchart)

Preconditions:

1. The employee should be authenticated by the portal and have the privileges to manage his / her own vacation time.
2. System already returned and displayed the home page.

Manager Handles Employee Request

Preconditions:

- The manger received an email about a new request and clicked the embedded link in the email.
- The manager is authenticated by the portal to manage his personals requests.
- The VTS displayed the separate section listing requests pending approval.
- The manager selected individual request.

Pseudocode For New Request Scenario

```
Function void createReq(Request: request){

        Boolean valid = validateRequest(request);

        If(valid){

                request.status=”pending approval”;

                saveRequest(request); //DB

                async notifyManager(request);

        }else{

                Handle Errors

        }

}

Function Boolean validateRequest(Request: req){

        //validate date from, to;

        validateReqFromAndTo(req.from, req.to);

        //validate vacation category balance

        validateCategoryBalance(req.categoryID, empID, req.balance)

}

Function Boolean validateReqFromAndTo(from, to){}

Function validateCategoryBalance(categoryID, empID, reqBalance){

        Int empBlanace=getEmployeeBalance(empID, categoryID);

        Return empBalance > reqBalance;

}

// The getEmployeeBalance function is an API integration with the HR legacy System
```

Pseudocode For New Request Approval Scenario

Preconditions: Manager is authenticated by the portal and reviewed the request.

```
Function changeRequestStatus(managerSelection, request){

        Request.status=managerSelection;

        manageRequest(request);

}

Function manageRequest(Request: request){

        Async notifyEmployee(request);

        updateRequest(request); //DB


}
```