**Contents**

1.0 Document Purpose

2.0 Intended Audience

3.0 Project Background, Objective(s)

3.1 Project Background

3.2 Project Objective

4.0 Design Pattern

5.0 Solution Diagram

6.0 Architecture Diagram

7.0 Flow Diagram

8.0 Use case Diagram

9.0 Class Diagram

10.0 E-R Diagram

11.0 User Requirements

11.1 Hardware

11.2 Software

12.0 Developer Requirements

12.1 Hardware

12.3 Technology

13.0 Solution Steps

14.0 Classes/function

15.0 Data Model

16.0 HTTP Status Code

# Document Purpose

This document describes the solution architecture for Roll Off form

# Intended Audience

This document is intended as a reference for the following roles and stakeholders who are interested to perform tasks in web application.

|  |  |
| --- | --- |
| Role | Nature of Engagement in WB Classics Portal Technical Architecture |
| Accounts team | Accounts can view details, roll off and employee list and fill feedback form |
| PSP Team | Psp team can see the rolloff request and can approved it or reject it. |
| Super Admin | Admin can see the requests when there is critical case like accounts want roll off but psp team don’t want. |

# Project Background, Objective(s)

## **Project Background**

Developed a project which helps to Roll Off the employee. This project provides the single page application due to this direct communication will be possible.

## **Project Objective**

## Management of company is facing problem while Roll Off of employees. In early days it was happening only through mail so there was a chance of miscommunication. To improve this scenario there should be a common platform where this process can work smoothly

# Design Pattern

|  |  |  |
| --- | --- | --- |
| # | Name | Description |
| 1 | ANGULAR | Angular is used for the frontend and designing of the application |
| 2 | API | API is used to process all the http  Requests. |
| 3 | SQL SERVER | SQL server is used to store data to the database |

**5.0. Solution Diagram:**

A picture containing letter

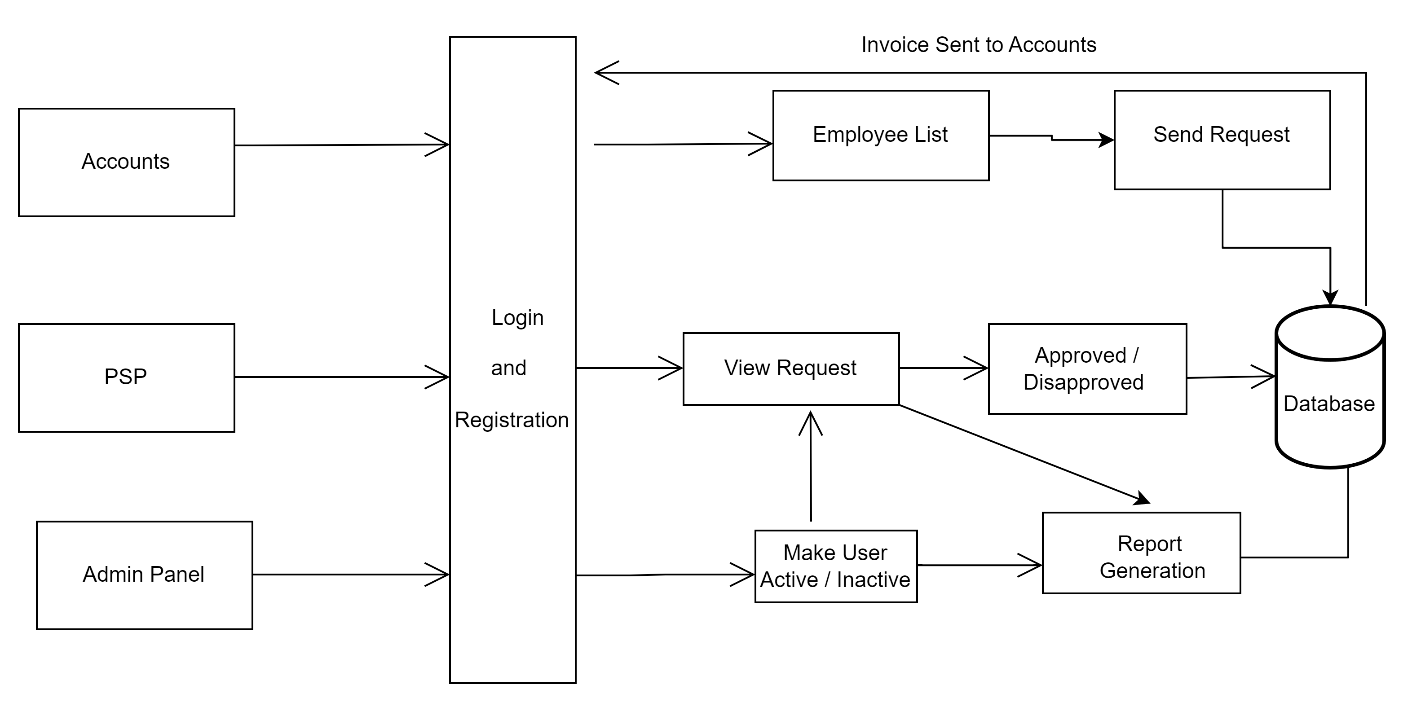
Description automatically generated

**6.0 Architecture Diagram:**

Diagram

Description automatically generated

**7.0 Flow Diagram:**



**8.0 Use case Diagram:**

Graphical user interface, application

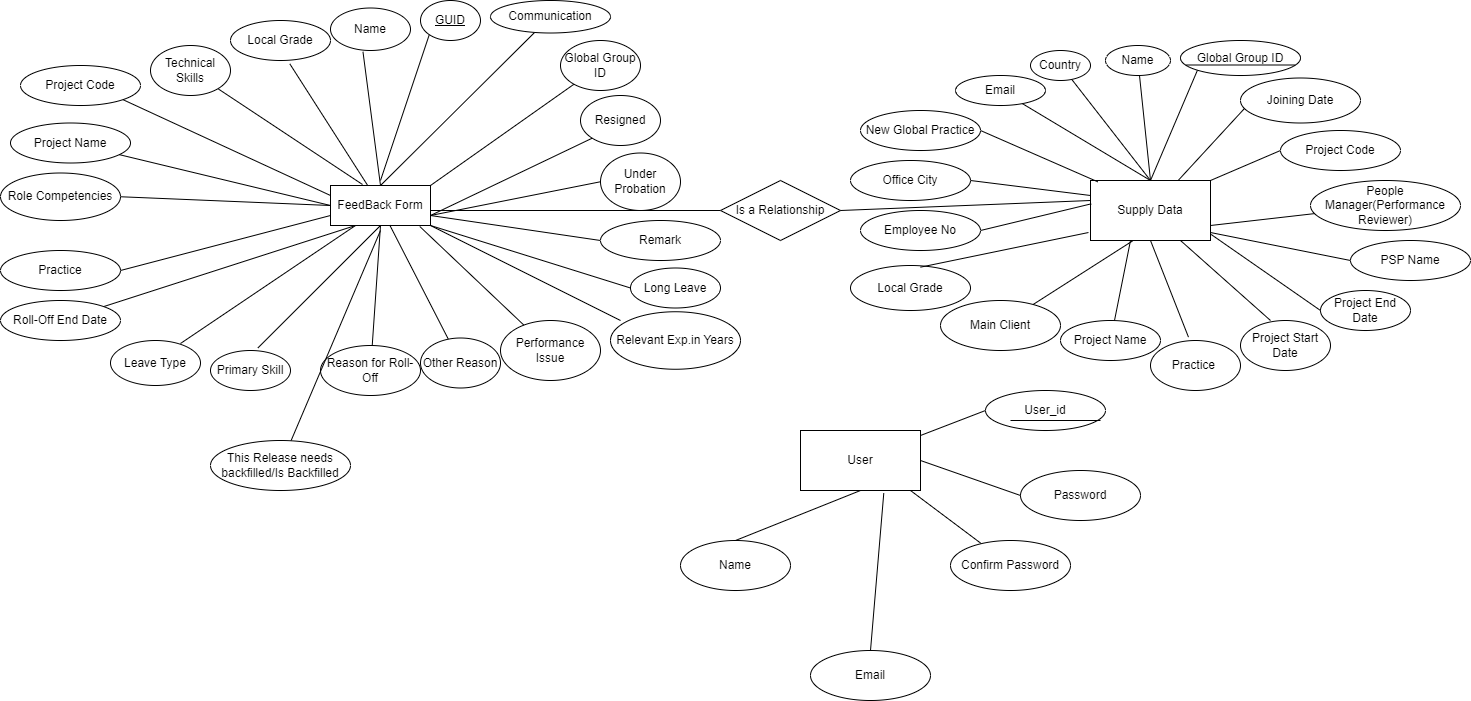
Description automatically generated

**9.0. Class Diagram:**

Graphical user interface

Description automatically generated

**10.0 ER Diagram:**



# 11.0 User Requirements

# 11.1 Hardware

# • OS: Windows.

# 11.2 Software

# • Any Latest Browsers.

# 12.0 Developer Requirements

# 12.1 Hardware

# • OS: Windows.

# 12.2 Software

# • Visual studio 2022.

# •SQL server management studio (ssms).

# • Node, Angular.

# 13.0 Solution Steps:

**13.1 Registration and Login-**

* Account and PSP will select themselves as Admin
* Admin will enter their username, name, email, password and confirm password.
* The input validation will be done

1. If validation fails, then it will return the error code and error description. with status code
2. If validation is successful, then the User details is stored in the database and success code is sent.

**13.2 Employee Details-**

* After registration and login successfully, Admin can view the employee details.
* Employee details will be available through excel sheet and details will be stored in database.
* Details like Global Group Id, Employee Number, Name, Local Grade, Main Client, Email, Joining Date, Project Code, Project Name, Project Start Date, Project End Date, People Manager etc.

**13.3 Post Feedback form/Send Roll Off Request-**

* Accounts team (admin) will start the Roll Off request by filling the Roll Off form of employee and send the request to the PSP team.

**13.4 View Roll Off Request and Feedback Data-**

* PSP team will view the request and approved / disapproved the request by verifying the given reasons of roll off.

# 14.0 Classes/function

|  |  |  |
| --- | --- | --- |
| **#** | **Class** | **Description** |
| 1 | RollOffTable.cs | Model holds the employee details |
| 2 | RollOffRepository.cs | It contains the methods to get create, update and delete roll off table details. |
| 3 | RollOffController.cs | It contains the core business logic for the creation, delete, updating and retrieval of Employee. Which calls the RollOffRepository class to manipulate the employee in database |
| 4 | FormTable.cs | This class deals with data accessibility for employee details. |
| 5 | FormRepository.cs | It contains the methods to get create,update and delete rolloff table details. |
| 6 | FormController.cs | It contains the core business logic for the adding the creation,delete and update and retrieval of employee which calls the FormRepository class to manipulate the employee in database |
| 7 | AdminTable.cs | It contains data creation and update. |
| 8 | AdminRepository .cs | It contains the methods to get create and update rolloff table details. |
| 9 | AdminController .cs | It contains the core business logic for the adding the creation and update and retrieval of employee which calls the AdminRepository class to manipulate the employee in database |
| 10 | RollOffContext.cs | It contains the context to the tables in the database that is rolloff table,admin table,form table. |

**15.0 Data Model:**

Graphical user interface, application

Description automatically generated

FK

# 15.0 API Canvas

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| POC | Path | Verb | API Description | Role | Auth |
| RollOff | /Api/RollOff | GET | To Get employee details | Admin | True |
| RollOff | /Api/RollOff | POST | To create employee list | Admin | True |
| RollOff | /Api/ RollOff/{ggid} | GET | Search employee by GGID | Admin | True |
| RollOff | /Api/ RollOff/{ggid} | PUT | Update employee by GGID | Admin | True |
| RollOff | /Api/ RollOff/{ggid} | DELETE | Delete employee by GGID | Admin | True |
| RollOff | /Api/ RollOff/employeeNo | GET | Search employee by EmployeeNo. | Admin | True |
| RollOff | /Api/ RollOff/{email} | GET | Search employee by Email | Admin | True |
| Form | /Api/Form | POST | Fill the feedback form | Admin | True |
| Form | /Api/Form | GET | To get the RollOff employee details | Admin | True |
| Form | /Api/Form/{ggid} | DELETE | Delete the employee details by GGID. | Admin | True |
| Form | /Api/Form/{ggid} | PUT | Update the employee details by GGID. | Admin | True |
| Admin | /Api/Admin | GET | User registration | Admin | True |
| Admin | /Api/Admin | POST | Get the registered user. | Admin | True |

# 16.0 HTTP Status Code

201 – Account registered, PSP registered.

200 - Request succeeded

400 – Inputs are invalid

404 – Data not found

502 – Bad gateway