# Expense Management System

The Expense Management System is designed to help organizations track and manage their expenses efficiently. The system is developed using a Monolithic architecture to ensure scalability, flexibility, and maintainability. The application is divided into multiple services, each responsible for a specific aspect of expense management.

## Monolithic architecture

The system is divided into the following modules:

1. **User Module**
2. **Expense Module**
3. **Category Module**
4. **Report Module**
5. **Notification Module**

## Entities and Their Relationships

### 1. User Module

- **User**
    - UserID (Primary Key)
    - Name
    - Email
    - Role (Admin, User)
    - Password (hashed)

**Relationships**:

- Each user can have multiple expenses.
- Users can have different roles affecting their permissions within the system.

### 2. Expense Module

- **Expense**
    - ExpenseID (Primary Key)
    - UserID (Foreign Key)
    - Amount
    - Date
    - CategoryID (Foreign Key)
    - Description

**Relationships**:

- Each expense is linked to a user.
- Each expense falls under a specific category.

### 3. Category Module

- **Category**
  - CategoryID (Primary Key)
  - Name
  - Description

**Relationships**:

- Categories are used to classify expenses.
- Categories are predefined and can be managed by admin users.

### 4. Report Module

- **Report**
  - ReportID (Primary Key)
  - UserID (Foreign Key)
  - Period (Month, Year)
  - TotalAmount
  - GeneratedDate

**Relationships**:

- Reports are generated for users based on their expenses.

### 5. Notification Module

- **Notification**
  - NotificationID (Primary Key)
  - UserID (Foreign Key)
  - Message
  - Date
  - Status (Read, Unread)

**Relationships**:

- Notifications are sent to users for various events (e.g., expense submission, report generation).

## Workflow

1. **User Registration and Authentication**:
   - Users register via the User Service.
   - Authentication and Authorization Service handles login and token generation.
2. **Expense Management**:
   - Users submit expenses through the Expense Service.
   - Each expense is categorized using the Category Service.
3. **Reporting**:
   - Users can generate reports of their expenses using the Report Service.

    - Reports summarize expenses over a specified period.
4. **Notifications**:
    - Users receive notifications for various events via the Notification Service.
    - Events include expense submissions and report generations.
5. **Admin Capabilities**:
    - Admin users can manage categories and oversee user activities.

Note: Make sure your project is having implementation of  following concepts:

1. Angular/ react as frontend technology
2. Spring rest Backend tecnology
3. Spring Data JPA
4. Swagger
5. Mockito, Junit
6. Jacoco for code coverage
7. Java 8 features
8. Exception Handling, Validation and Logs
9. Push code to git