**Báo cáo bài tập**

**Đề tài:** Quản trị nhân viên.

**Mô tả:** Ứng dụng quản trị nhân viên, user có thể đăng ký và đăng nhập. Khi user đăng nhập hệ thống sẽ kiểm tra xem tài khoản đó còn hoạt động hay không (isActive) nếu hoạt động thì hiển thị danh sách thông tin về nhân viên. User có thể thêm, sửa, xóa thông tin nhân viên.

Thông tin user: ID, full name, email, password, role, isActive, isDelete.

Thông tin nhân viên (employee): ID, first name, last name, email, password, dob (date of birth), isActive, isDelete.

Link code: https://github.com/daitc2209/EmployeeManagement

***Sơ đồ Use case tổng quát:***

Diagram

Description automatically generated

***Sơ đồ Sequence:***

1. Đăng nhập:

Graphical user interface, diagram, application

Description automatically generated

1. Đăng ký:

Chart, diagram, box and whisker chart

Description automatically generated with medium confidence

1. Thêm user:

Diagram

Description automatically generated

1. Sửa user:

Diagram

Description automatically generated with low confidence

1. Hiển thị và xóa user:

A picture containing graphical user interface

Description automatically generated

1. Tìm kiếm user:

Diagram

Description automatically generated

1. Sửa nhân viên:

A picture containing diagram

Description automatically generated

1. Hiển thị và xóa nhân viên:

Diagram

Description automatically generated

1. Tìm kiếm nhân viên:

Diagram

Description automatically generated with medium confidence

***Biểu đồ Activity:***

1. Đăng nhập

Diagram

Description automatically generated

1. Đăng ký

Diagram

Description automatically generated

1. Thêm user

Diagram

Description automatically generated

1. Sửa user

Diagram

Description automatically generated

1. Xóa user

Diagram

Description automatically generated

1. Sửa nhân viên

Diagram

Description automatically generated

1. Xóa nhân viên

Diagram

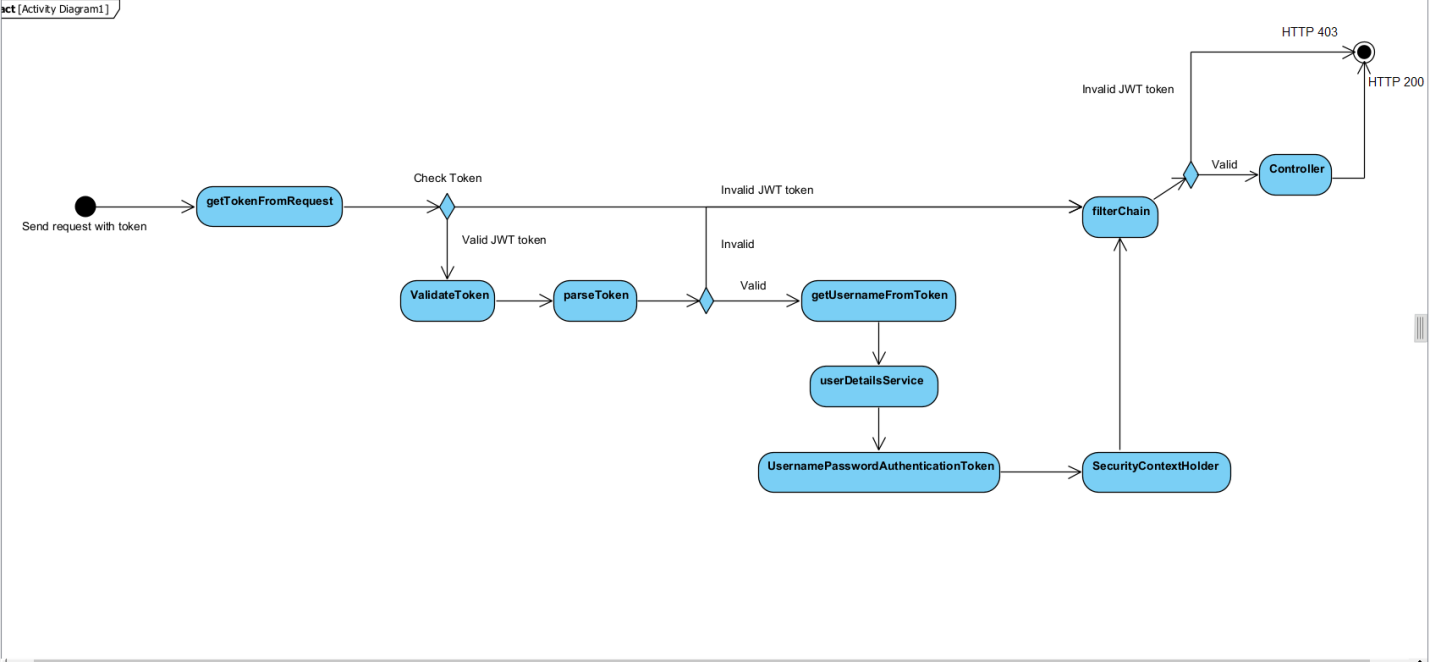
Description automatically generated

1. Tìm kiếm:

Diagram

Description automatically generated

**Mô tả luồng JWT:**



Nhận 1 request với token, đi vào hàm getTokenFromRequest nó trích xuất Header xác thực từ request và kiểm tra xem nó có bắt đầu bằng Bearer hay không, nếu có sẽ trả về phần còn lại của Header token. Nếu không thì sẽ trả về null và đi tới phương thức filterChain để tiếp tục quá trình request mà không cần xác thực và trả về HTTP 403.

Sau đó đi vào hàm ValidateToken để gọi phương thức parseToken. Trong hàm parseToken này sẽ giải mã Token nếu không có lỗi, hàm sẽ trả về đối tượng chứa các thông tin về người dùng. Nếu không sẽ trả về null.

Bộ lọc sẽ lấy tên người dùng từ hàm getUsernameFromToken và sử dụng userDetailsService để tải thông tin chi tiết người dùng. Và tạo một đối tượng authentication bằng cách sử dụng đối tượng UsernamePasswordAuthenticationToken và đặt nó vào context bảo mật của Spring cho phép quá trình xử lý request tiếp tục.

filterChain cho phép tiếp tục xử lý các request vào Controller và trả về HTTP 200 nếu như đã được xác thực.

***Biểu đồ mô hình thực thể Erd:***

Diagram

Description automatically generated

**Database:**

Table employee:

|  |  |  |
| --- | --- | --- |
| **Name Field** | **Types** | **Constraint** |
| emp\_id | Int | Primary key, identity |
| first\_name | varchar(45) | Not null |
| last\_name | Varchar(20) | Not null |
| email\_id | Varchar(45) | unique,not null |
| password | Varchar(255) | Not null |
| dob | date | Not null |
| address | Varchar(100) | Not null |
| is\_delete | boolean |  |
| is\_active | boolean |  |

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text

Description automatically generated

Table user:

|  |  |  |
| --- | --- | --- |
| **Name Field** | **Types** | **Constraint** |
| user\_id | Int | Primary key, identity |
| email | varchar(45) | Not null, unique |
| password | Varchar(255) | Not null |
| name | Varchar(45) | not null |
| is\_delete | boolean |  |
| is\_active | boolean |  |
| role | Varchar(25) | Not null |

Text, application

Description automatically generated

Graphical user interface, text

Description automatically generated