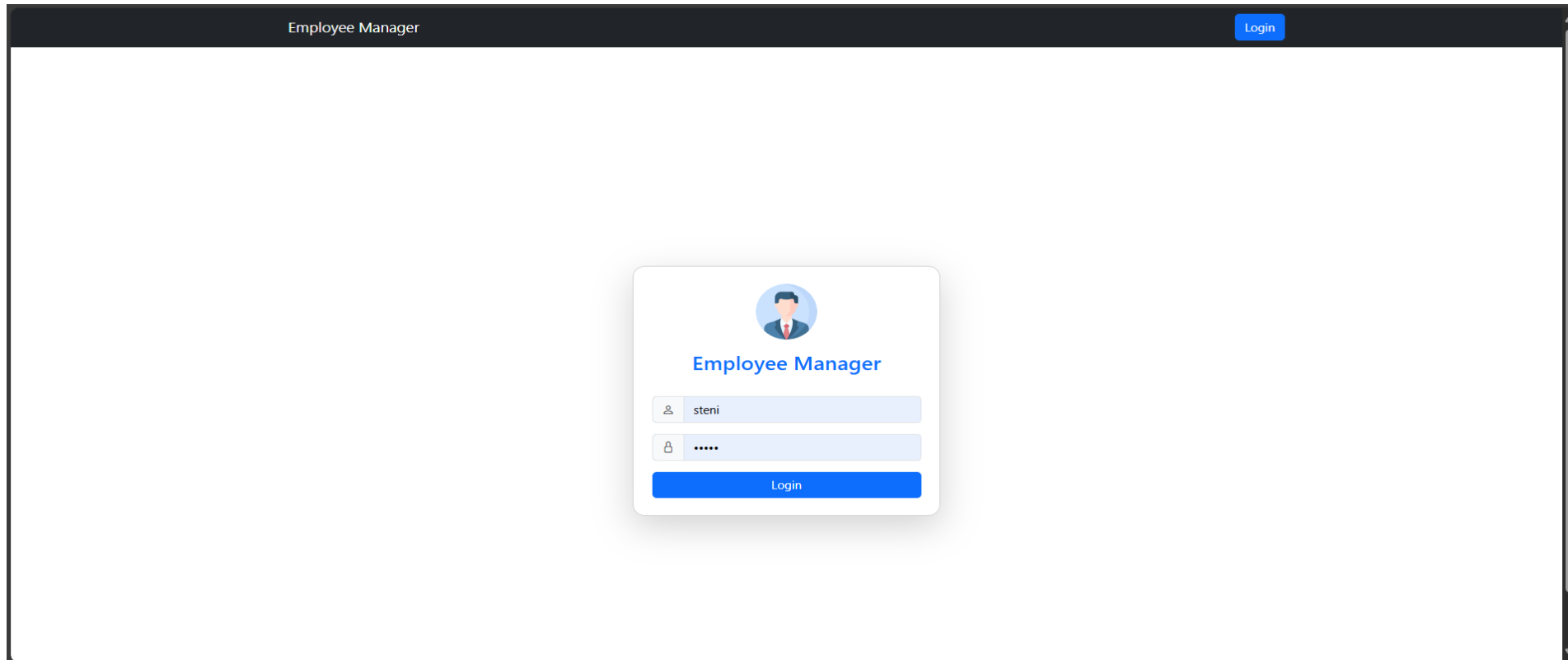


# Employee Manager

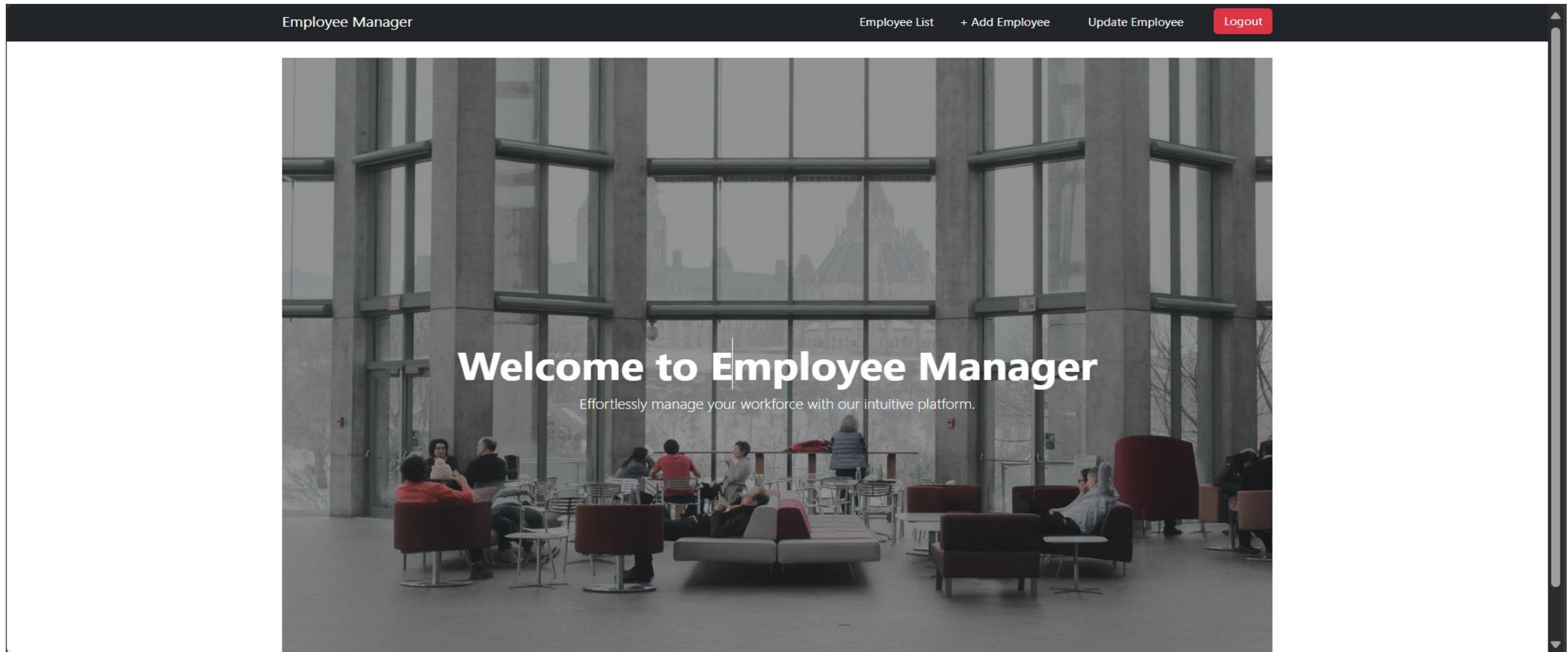
- Login Page:



- Only after Login the header contents will be fully shown
- Prevents Unauthorized Users from Viewing the Dashboard

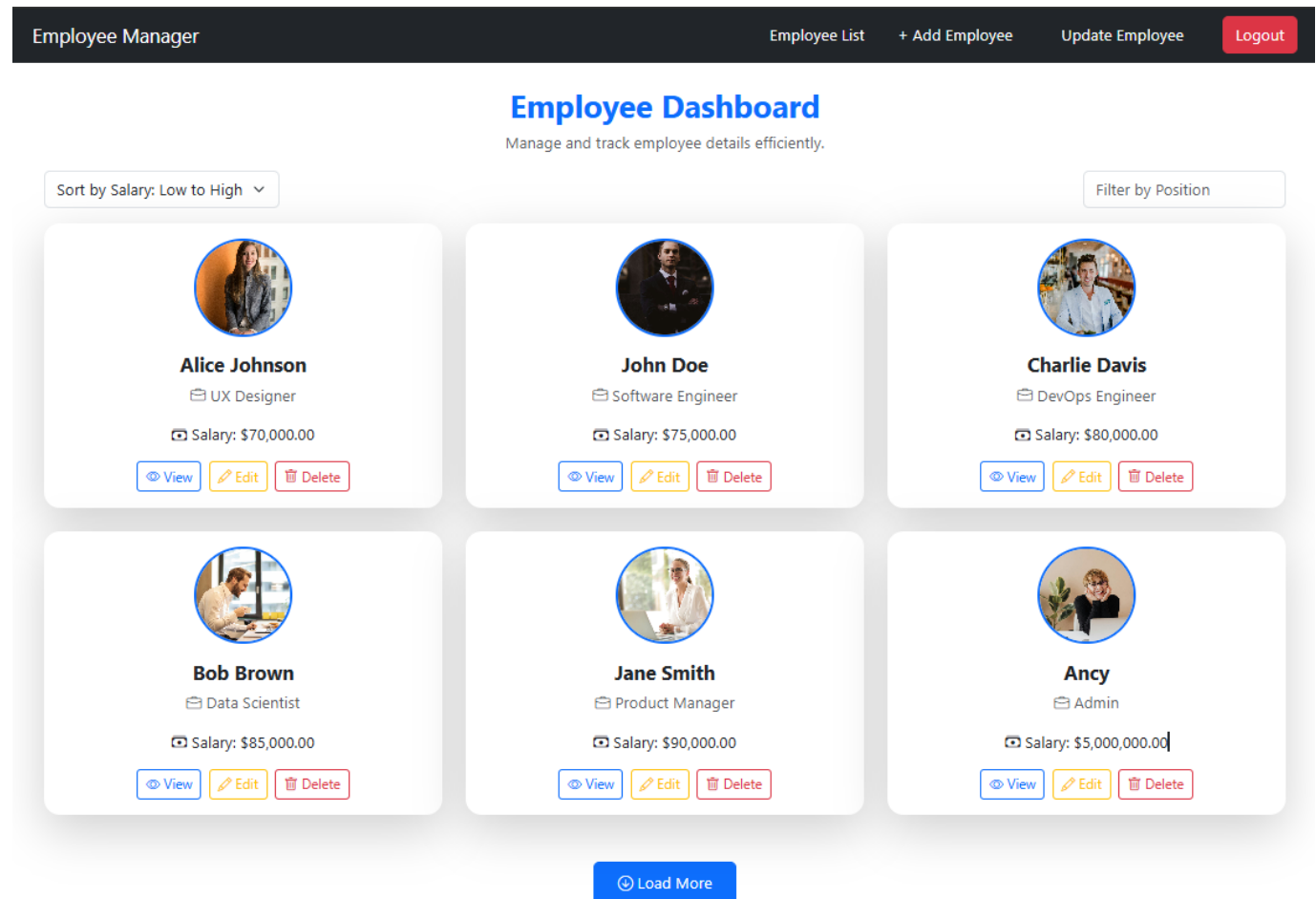
# Home Page :

- Once Login it directs to Home Page and all the fields of header is visible
- Home component is used



# Employee list

- Displays Employee List in a modern card layoutWhere the details of all the employees are shown in the form of cards where I implemented Custom and built-in pipes for
- Shows Employee Name, Position, and Salary  
Profile pictures dynamically fetched from API  
Sort Employees by Salary (Ascending/Descending)  
Filter Employees by Position
- Employee List -Component



# Sorting & Filtering

Sort Employees by Salary

Filter Employees by Job Role

Uses Angular Pipes for real-time sorting & filtering

**Pipes Are used**

Sort high to low :


Employee Manager

Employee List+ Add EmployeeUpdate EmployeeLogout


## Employee Dashboard

Manage and track employee details efficiently.


Sort by Salary: High to Low ▾Filter by Position




**Ancy**  
Admin  
Salary: \$5,000,000.00  
[View](#) [Edit](#) [Delete](#)




**Jane Smith**  
Product Manager  
Salary: \$90,000.00  
[View](#) [Edit](#) [Delete](#)




**Bob Brown**  
Data Scientist  
Salary: \$85,000.00  
[View](#) [Edit](#) [Delete](#)



**Charlie Davis**  
DevOps Engineer  
Salary: \$80,000.00  
[View](#) [Edit](#) [Delete](#)



**John Doe**  
Software Engineer  
Salary: \$75,000.00  
[View](#) [Edit](#) [Delete](#)



**Alice Johnson**  
UX Designer  
Salary: \$70,000.00  
[View](#) [Edit](#) [Delete](#)

# Sort by position:

Employee Manager


Employee List+ Add EmployeeUpdate EmployeeLogout

Employee Dashboard

Manage and track employee details efficiently.

Sort by Salary: High to Low

Admin



Ancy

Admin

Salary: \$5,000,000.00

View

Edit

Delete

Load More

# Currency symbol in salary built-in filter :

## Employee Dashboard

Manage and track employee details efficiently.

Sort by Salary: High to Low ▾

Filter by Position



**Ancy**

Admin

Salary: \$5,000,000.00

View

Edit

Delete



**Jane Smith**

Product Manager

Salary: \$90,000.00

View

Edit

Delete



**Bob Brown**

Data Scientist

Salary: \$85,000.00

View

Edit

Delete



**Charlie Davis**

DevOps Engineer

Salary: \$80,000.00

View

Edit

Delete



**John Doe**

Software Engineer

Salary: \$75,000.00

View

Edit

Delete



**Alice Johnson**

UX Designer

Salary: \$70,000.00

View

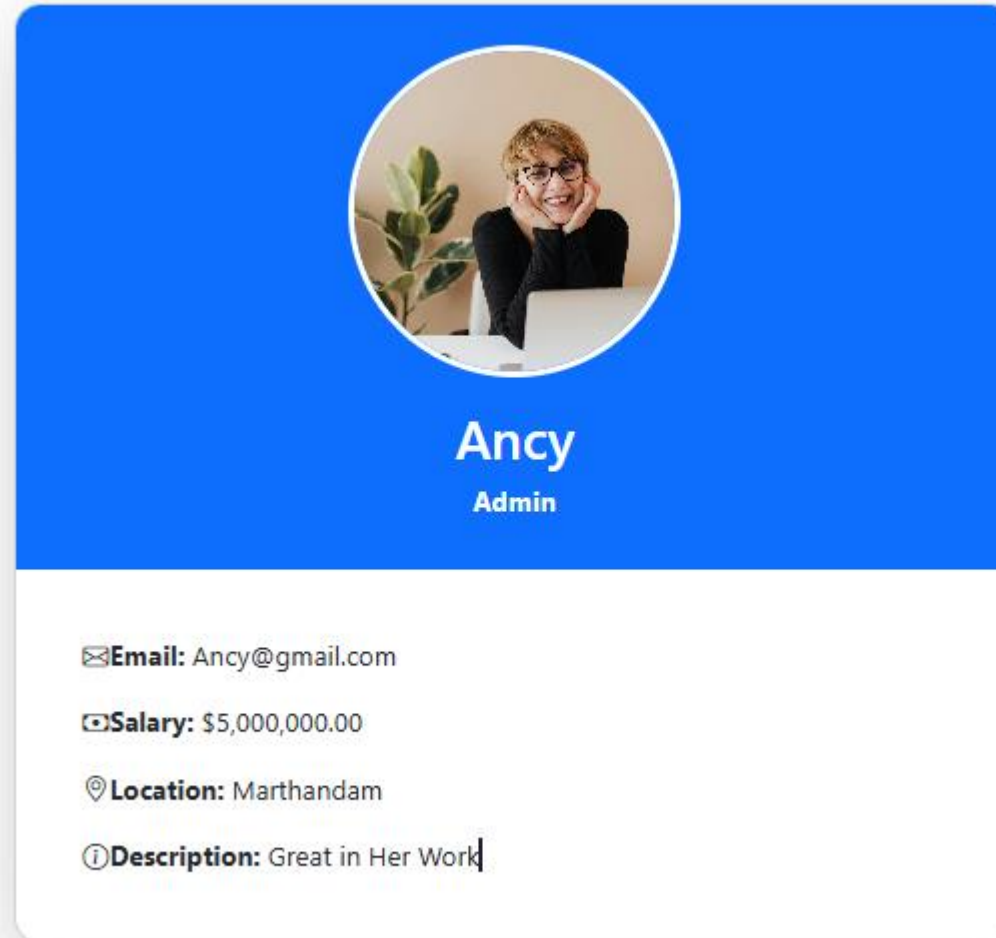
Edit

Delete

Load More

## View in Employee card:

- Employee Detail –component is used
- shows the detailed details of employee



## Edit in Employee Dashbord:

Employee form component is used

It uses Model base forms

Delete to delete the employee data from DB

Employee Manager

[Employee List](#)

[+ Add Employee](#)

[Update Employee](#)

[Logout](#)

### Edit Employee

Name

Ancy

Position

Admin

Email

Ancy@gmail.com

Salary

5000000

[Save Employee](#)

[Cancel](#)



# Add Employee:

To add a new employee to the DB

Employee Manager

Employee List

+ Add Employee

Update Employee

Logout

Add Employee

Name

Position

Email

Salary

Save Employee

Cancel

It is similar to edit

# Update Employee:

Template form component is used for the template

We use Template based form

Employee Manager

Employee List + Add Employee

Update Employee

Logout

Edit Employee

Select Employee

Ancy

Name

Ancy

Position

Admin

Email

Ancy@gmail.com

Salary

5000000

Location

Marthandam

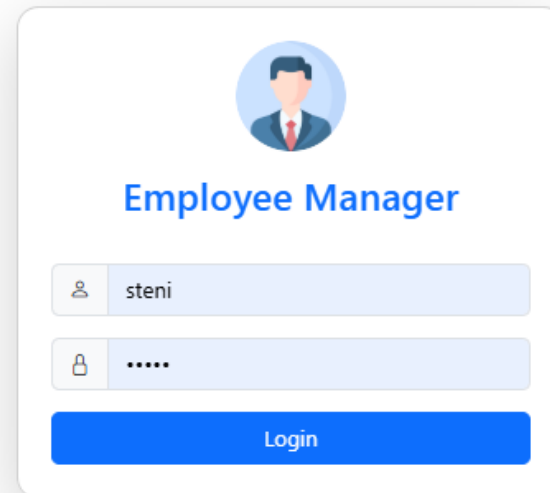
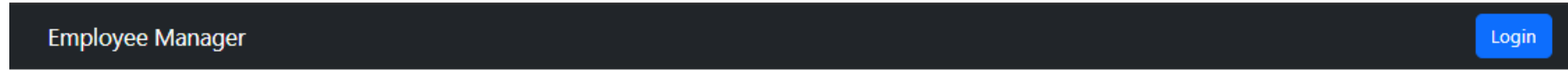
Description

Great in Her Work

Update Employee

# Logout:

By clicking logout it directs to login page.



## Authentication & Authorization

- Where:** login.component.ts, auth.guard.ts, app.component.ts
- When:** During login, logout, and route access
- Purpose:** Ensures only authenticated users access the dashboard

## Employee CRUD (Create, Read, Update, Delete)

- Where:** employee-list.component.ts, employee-form.component.ts, employee.service.ts
- When:** When adding, editing, viewing, or deleting employees
- Purpose:** Manages employee data using API calls

## Routing & Navigation

- Where:** app-routing.module.ts
- Purpose:** Handles navigation between different sections
- When:** When navigating between pages (/employees, /edit/:id, /login)

## Sorting & Filtering

- Where:** sortSalary.pipe.ts, filterPosition.pipe.ts, employee-list.component.html
- When:** When sorting salaries or filtering employees by position
- Purpose:** Improves employee search and organization

## API & Data Management

- Where:** employee.service.ts, db.json
- When:** When fetching, adding, updating, or deleting employee data
- Purpose:** Handles data communication with the backend

## Secure Route Protection (AuthGuard)

- Where:** auth.guard.ts, app-routing.module.ts
- When:** When an unauthorized user tries to access protected routes
- Purpose:** Redirects users to login if not authenticated

## Form Handling (Template & Reactive Forms)

- Where:** employee-form.component.ts
- When:** When submitting employee data via forms
- Purpose:** Validates and processes employee input

## Responsive UI & Modern Design

- Where:** styles.css, Bootstrap, employee-list.component.html
- When:** When displaying the application on different screen sizes
- Purpose:** Ensures a clean, user-friendly interface

## Load More Feature

- Where:** employee-list.component.ts, employee-list.component.html
- When:** When displaying a large list of employees
- Purpose:** Loads more employees dynamically for better performance