Employee Management System - API Documentation

# Table of Contents:

1. API Endpoints  
 - Employee Management APIs  
 - Salary & Attendance APIs  
 - Department Statistics API  
2. Authentication and Authorization  
3. How to Run the Project  
 - Prerequisites  
 - Steps to Run  
 - Accessing Swagger UI for API testing

# 1. API Endpoints

## Employee Management APIs

These APIs enable administrators to create, update, delete, and retrieve employee records.

### Create Employee

• URL: POST http://localhost:9090 /employees  
• Access: Admin only (ADMIN role)  
• Request Body:

{  
 "name": "Krishna",  
 "email": "Krishna@gmail.com",  
 "password": "Krishna@123",  
 "dateOfBirth": "1990-01-01",  
 "department": "IT",  
 "salary": 45000,  
 "city": "Pune",  
 "roles": ["USER"]  
}

• Response (201 Created):

{  
 "id": 1,  
 "name": "Krishna",  
 "email": "Krishna@gmail.com",  
 "password": "Krishna@123",  
 "dateOfBirth": "1990-01-01",  
 "department": "IT",  
 "salary": 45000,  
 "city": "Pune",  
 "roles": ["USER"]

}

### Retrieve All Employees

• URL: GET http://localhost:9090/employees  
• Access: Admin & User (ADMIN, USER roles)  
• Response (200 OK):

### [

### {

### "id": 5,

### "name": "Jaywant Mhala",

### "email": "jay@gmail.com",

### "password": "$2a$10$z8qmJETWQBhem9dsczGom.6G8NzIyRqZAz1IMNw./mtnhmoNJDuFi",

### "dateOfBirth": "2020-11-12",

### "department": "IT",

### "salary": 50000.0,

### "city": "Chandur",

### "roles": [

### "USER",

### "ADMIN"

### ],

### "attendance": 12,

### "enabled": true,

### "credentialsNonExpired": true,

### "accountNonExpired": true,

### "username": "jay@gmail.com",

### "accountNonLocked": true,

### "authorities": [

### {

### "authority": "USER"

### },

### {

### "authority": "ADMIN"

### }

### ]

### },

### {

### "id": 10,

### "name": "Piyush Mhala",

### "email": "piyush@gmail.com",

### "password": "$2a$10$XPwOGmIzEcbQycsk6gJrhuePT/VkUMHyv1Xr385FF/.V7qCZqaWV2",

### "dateOfBirth": "2020-11-12",

### "department": "IT",

### "salary": 12112.0,

### "city": "Chandur",

### "roles": [

### "USER"

### ],

### "attendance": 2,

### "enabled": true,

### "credentialsNonExpired": true,

### "accountNonExpired": true,

### "username": "piyush@gmail.com",

### "accountNonLocked": true,

### "authorities": [

### {

### "authority": "USER"

### }

### ]

### },

### {

### "id": 21,

### "name": "Rahul Kumar",

### "email": "rahul@gmail.com",

### "password": "$2a$10$C2SFFnivHDPzI9FQQfkqV.L2QK420n1jyQujYXsormeGdcCJBWKO.",

### "dateOfBirth": "1990-01-01",

### "department": "HR",

### "salary": 75000.0,

### "city": "New York",

### "roles": [

### "USER",

### "ADMIN"

### ],

### "attendance": 20,

### "enabled": true,

### "credentialsNonExpired": true,

### "accountNonExpired": true,

### "username": "rahul@gmail.com",

### "accountNonLocked": true,

### "authorities": [

### {

### "authority": "USER"

### },

### {

### "authority": "ADMIN"

### }

### ]

### },

### {

### "id": 22,

### "name": "Vishal",

### "email": "vishal@gmail.com",

### "password": "$2a$10$hzfum9t2hpjUt84Xx6wlbexRe0.ft5UYD82317S5hY6iqi173Jmh2",

### "dateOfBirth": "2000-01-01",

### "department": "MANAGER",

### "salary": 15000.0,

### "city": "PATNA",

### "roles": [

### "USER"

### ],

### "attendance": 20,

### "enabled": true,

### "credentialsNonExpired": true,

### "accountNonExpired": true,

### "username": "vishal@gmail.com",

### "accountNonLocked": true,

### "authorities": [

### {

### "authority": "USER"

### }

### ]

### },

### {

### "id": 23,

### "name": "Abhi",

### "email": "abbi@gmail.com",

### "password": "$2a$10$p/qWXR331cbFIAR9jb7nZeFnYa0xUMck3kAXRyxMs9GiGKfo518Kq",

### "dateOfBirth": "2000-01-01",

### "department": "MANAGER",

### "salary": 15000.0,

### "city": "PATNA",

### "roles": [

### "USER"

### ],

### "attendance": 20,

### "enabled": true,

### "credentialsNonExpired": true,

### "accountNonExpired": true,

### "username": "abbi@gmail.com",

### "accountNonLocked": true,

### "authorities": [

### {

### "authority": "USER"

### }

### ]

### }

### ]

### Retrieve Employee by ID

• URL: GET http://localhost:9090/employees/{id}  
• Access: Admin & User (ADMIN, USER roles)  
• Response (200 OK):

### {

### "id": 5,

### "name": "Jaywant Mhala",

### "email": "jay@gmail.com",

### "password": "$2a$10$z8qmJETWQBhem9dsczGom.6G8NzIyRqZAz1IMNw./mtnhmoNJDuFi",

### "dateOfBirth": "2020-11-12",

### "department": "IT",

### "salary": 50000.0,

### "city": "Chandur",

### "roles": [

### "USER",

### "ADMIN"

### ],

### "attendance": 12,

### "enabled": true,

### "credentialsNonExpired": true,

### "accountNonExpired": true,

### "username": "jay@gmail.com",

### "accountNonLocked": true,

### "authorities": [

### {

### "authority": "USER"

### },

### {

### "authority": "ADMIN"

### }

### ]

### }

### Update Employee

• URL: PUT /employees/{id}  
• Access: Admin only (ADMIN role)  
• Request Body:

{  
 "name": "Jaywant",  
 "email": "jaywant@gmail.com",  
 "city": "Pune"  
}

• Response (200 OK):

{  
 "id": 1,  
 "name": "Jaywant Mhala",  
 "email": "jaywant@gmail.com”,  
 "city": "Pune",  
 ...  
}

### Delete Employee

• URL: DELETE http://localhost:9090/employees/{id}  
• Access: Admin only (ADMIN role)  
• Response (200 OK):

{  
 "message": "Employee deleted successfully !!"  
}

## Salary & Attendance APIs

### View Employee Salaries

• URL: GET http://localhost:9090/employees/salaries  
• Access: Admin only (ADMIN role)  
• Response (200 OK):

[  
 {  
 "name": "Jaywant Mhala",

"salary": 50000.0  
 },  
 {  
"name": "Rahul Kumar",

"salary": 75000.0  
 }  
]

### View Employee Attendance

• URL: GET http://localhost:9090/employees/attendance  
• Access: Admin only (ADMIN role)  
• Response (200 OK):

[  
 {  
"name": "Jaywant Mhala",

"attendance": 12  
 },  
 {  
"name": "Rahul Kumar",

"attendance": 20  
 }  
]

## Department Statistics API

### Get Department-wise Employee Count

• URL: GET http://localhost:9090/employees/department-employee-count  
• Access: Admin only (ADMIN role)  
• Response (200 OK):

[  
 {  
"Department Name": "IT",

"Employee Count": 2  
 },  
 {  
 "Department Name": "MANAGER",

"Employee Count": 2  
 }  
]

# 2. Authentication and Authorization

\*\*Authentication\*\*: The Employee Management System uses \*\*Spring Security\*\* for user authentication. Users need to provide valid credentials (email and password) to access the system.  
  
\*\*Authorization\*\*:  
• \*\*ADMIN\*\*: Has full access to all CRUD operations, salary, attendance, and department statistics.  
• \*\*USER\*\*: Has limited access to view employee details (GET requests only).  
  
Authentication is managed using \*\*Basic Authentication\*\*.

# 3. How to Run the Project

## Prerequisites

1. \*\*Java\*\*: Ensure you have JDK 17 or later installed.  
2. \*\*Maven\*\*: Install Maven for dependency management.  
3. \*\*MySQL/PostgreSQL\*\*: Set up a database server.  
4. \*\*Postman/Swagger\*\*: Tools for testing APIs.

1. \*\*Configure the Database\*\*  
 Open the `application.properties` file and configure the database settings:  
 ```properties  
 spring.datasource.url=jdbc:mysql://localhost:3306/Xenosis  
 spring.datasource.username=root  
 spring.datasource.password=root  
   
2. \*\*Build and Run the Project\*\*  
 In the project root directory, run the following Maven commands:  
 ```bash  
 mvn clean install  
 mvn spring-boot:run  
 ```  
  
3. \*\*Access the APIs\*\*  
 The application will be accessible at:  
 ```bash  
 http://localhost:9090/employees  
 ```

## Accessing Swagger UI

You can test and explore all the available APIs through Swagger UI.  
  
1. Open your browser and go to:  
 ```bash  
 http://localhost:9090/swagger-ui.html  
 ```  
2. Use the interactive Swagger UI to test the API endpoints and view the API documentation.