**Attendance-Management**

# Features:

* Swipe In/Out.
* EOD total hours calculation for all employees.
* Attendance calculation based on total number of hours present in office.
* Notification of absenteeism (Desired).
* GraphQL API for fetching the attendance records.

# Services:

1. Gateway-Service.
2. Registry-Service.
3. Swipe-Service.
4. Attendance-Service.
5. Notification-Service.
6. Common-Service.

## Gateway-Service:

This is API Gateway of application. This is responsible to redirect all the requests to respective services. In future we can implement the security here and can integrate the authentication and authorization as well.

## Registry-Service:

This is a eureka registry service. All the services will be registered here, and we can monitor them.

## Swipe-Service:

This service is responsible to store the swipe in and swipe out events and will perform some calculation to find out the working hours of the employees. This service will also publish the attendance records to Kafka for attendance service.

## Attendance-Service:

This service will listen the data from Kafka to store the working and total hours of the employee in a day and then will provide the attendance reports. GraphQL is integrated with this service to enhance the endpoints.

## Notification-Service:

This service will listen the events from Kafka and is responsible to send the notification or mails to the employees.

## Common-Service:

This is not a web or rest service. This is just to have the common code that will be used in multiple services. This service is getting used as a dependency in other services.

# Databases:

* Swipe-Service is using the relational database that is MySQL.
* Attendance-Service is using MongoDB.

# Kafka-Topics:

* swipe-events-topic
* notification-events-topic

# Junit Test Cases:

Junit testcases are written for swipe-service. Code coverage is 95% as of now that can be found here: **swipe-service/target/coverage-reports/jacoco/index.html**

# GraphQL:

GraphiQL Url: <http://localhost:8765/attendance-service/graphiql?path=/attendance-service/graphql>

Postman: <http://localhost:8765/attendance-service/graphql?path=/graphql>

# API:

## Swipe-In:

URL: <http://localhost:8765/swipe-service/v1/swipe>

Request Method: POST

Request Body: {

    "userId":1,

    "swipeType":"IN"

}

## Swipe-out:

URL: <http://localhost:8765/swipe-service/v1/swipe>

Request Method: POST

Request Body: {

    "userId":1,

    "swipeType":"OUT"

}

## Manual Swipe-In:

URL: <http://localhost:8084/v1/swipe/manual>

Request Method: POST

Request Body: {

    "userId":3,

    "swipeType":"IN",

    "date":"2024-02-28",

    "swipeTime":"09:30:00"

}

## Manual Swipe-Out:

URL: <http://localhost:8084/v1/swipe/manual>

Request Method: POST

Request Body: {

    "userId":3,

    "swipeType":"OUT",

    "date":"2024-02-28",

    "swipeTime":" 23:40:00"

}

## Fetch APIs:

URL: [http://localhost:8765/attendance-service/v1/absent/<date](http://localhost:8765/attendance-service/v1/absent/%3cdate)>

URL: [http://localhost:8765/attendance-service/v1/attendance/<userId>?startDate=<startDate>&endDate=<endDate](http://localhost:8765/attendance-service/v1/attendance/%3cuserId%3e?startDate=%3cstartDate%3e&endDate=%3cendDate)>

URL: [http://localhost:8765/attendance-service/v1/attendance/<userId>/<attendanceStatus>?startDate=<startDate>&endDate=<endDate](http://localhost:8765/attendance-service/v1/attendance/%3cuserId%3e/%3cattendanceStatus%3e?startDate=%3cstartDate%3e&endDate=%3cendDate)>

# Note:

Fetch APIs upgraded using the GraphQL. Please check the GraphQL section to use them.