# API Documentation

used Swagger in this application. Once Application starts you can access Swagger API Doc at :

<http://localhost:8080/swagger-ui/index.html#/>

## User Authentication API Documentation

**User Endpoints**

**1. User Registration**

* **Endpoint:** POST /register
* **Description:** Register a new user in the system.
* **Request Body:**

{

"id": 0,

"username": "string",

"password": "string",

"email": "string"

}

* **Responses:200 OK**
  + - **Response Body:**

json

Copy code

{

"id": 0,

"username": "string",

"password": "string",

"email": "string"

}

* + **400 Bad Request**: Invalid input data (e.g., missing required fields).
  + **409 Conflict**: Username or email already exists.

**2. User Login**

* **Endpoint:** POST /login
* **Description:** Authenticate a user and log them into the system.
* **Request Body:**

{

"username": "string",

"password": "string"

}

* **Responses:200 OK**
  + - **Response Body:**

{

"string" Authentication token

}

* + **401 Unauthorized**: Invalid username or password.

## Health Metrics Endpoints

**1. Edit Health Metric**

* **Endpoint:** PUT /api/metrics/edit
* **Description:** Update an existing health metric.
* **Parameters:**
  + **id** *(query)*: The ID of the metric to be edited (required).
* **Request Body:**

{

"userId": 0,

"metricType": "Weight",

"value": 0,

"updatedDate": "2024-09-23",

"createdDate": "2024-09-23"

}

* **Responses:200 OK**

{

"id": 0,

"metricType": "Weight",

"value": 0,

"createdDate": "2024-09-23",

"user": {

"id": 0,

"username": "string"

}

}

**2. Add Health Metric**

* **Endpoint:** POST /api/metrics/add
* **Description:** Log a new health metric.
* **Request Body:**

{

"userId": 0,

"metricType": "Weight",

"value": 0,

"updatedDate": "2024-09-23",

"createdDate": "2024-09-23"

}

* **Responses:200 OK**

{

"id": 0,

"metricType": "Weight",

"value": 0,

"createdDate": "2024-09-23",

"user": {

"id": 0,

"username": "string"

}

}

**3. Get Health Metric**

* **Endpoint:** GET /api/metrics/getHealthMetric
* **Description:** Retrieve a specific health metric by ID.
* **Parameters:**
  + **id** *(query)*- The ID of the metric to retrieve (required).
* **Responses:200 OK**

{

"id": 0,

"metricType": "Weight",

"value": 0,

"createdDate": "2024-09-23",

"user": {

"id": 0,

"username": "string"

}

}

**4. Get All Health Metrics**

* **Endpoint:** GET /api/metrics/getAll
* **Description:** Retrieve all health metrics for a specific user.
* **Parameters:**
  + **userId** *(query)*- The ID of the user (required).
* **Responses:200 OK**

[

{

"id": 0,

"metricType": "Weight",

"value": 0,

"createdDate": "2024-09-23",

"user": {

"id": 0,

"username": "string"

}

}

]

## Health Goal Endpoints

**1. Edit Health Goal**

* **Endpoint:** PUT /api/healthGoal/edit
* **Description:** Update an existing health goal.
* **Parameters:**
  + **id** *(query)*: The ID of the goal to be edited (required).
* **Request Body:**

{

"userId": 1,

"targetValue": 250,

"startDate": "2024-09-22",

"endDate": "2024-09-22",

"isActive": true

}

* **Responses:200 OK**

{

"id": 1,

"goalType": "Weight\_Goal",

"targetValue": 250,

"achieved": false,

"startDate": "2024-09-22",

"endDate": "2024-09-22",

"createdAt": "2024-09-22",

"updatedAt": "2024-09-23",

"isActive": true,

"user": {

"id": 1,

"username": "Thushan"

}

}

**2. Track Progress**

* **Endpoint:** POST /api/healthGoal/users/track-progress
* **Description:** Track progress for a specific user’s goals.
* **Parameters:**
  + **userId** *(query)*- The ID of the user (required).
* **Responses:200 OK**

[

{

"id": 0,

"goalType": "Weight\_Goal",

"targetValue": 0,

"achieved": true,

"startDate": "2024-09-23",

"endDate": "2024-09-23",

"createdAt": "2024-09-23",

"updatedAt": "2024-09-23",

"isActive": true,

"user": {

"id": 0,

"username": "string"

}

}

]

**3. Add Health Goal**

* **Endpoint:** POST /api/healthGoal/add
* **Description:** Create a new health goal.
* **Request Body:**

{

"userId": 0,

"goalType": "Weight\_Goal",

"targetValue": 0,

"startDate": "2024-09-23",

"endDate": "2024-09-23",

"isActive": true,

"achieved": true

}

* **Responses:200 OK**

{

"id": 0,

"goalType": "Weight\_Goal",

"targetValue": 0,

"achieved": true,

"startDate": "2024-09-23",

"endDate": "2024-09-23",

"createdAt": "2024-09-23",

"updatedAt": "2024-09-23",

"isActive": true,

"user": {

"id": 0,

"username": "string"

}

}

**4. Get Health Goals**

* **Endpoint:** GET /api/healthGoal/get
* **Description:** Retrieve all health goals for a specific user.
* **Parameters:**
  + **userId** *(query)*- The ID of the user (required).
* **Responses:200 OK**

[

{

"id": 1,

"goalType": "Weight\_Goal",

"targetValue": 60,

"achieved": false,

"startDate": "2024-09-22",

"endDate": "2024-09-22",

"createdAt": "2024-09-22",

"updatedAt": "2024-09-22",

"isActive": false,

"user": {

"id": 1,

"username": "Thushan"

}

}

]

## Notification API Documentation

**Notification Endpoints**

**1. Send Goal Reminder**

* **Endpoint:** POST /api/notification/send-goal-reminder/
* **Description:** Send a reminder notification to a user about their health goals.
* **Parameters:**
  + **userId** *(query)*The ID of the user to whom the reminder will be sent (required).
* **Responses:200 OK**
  + - **Response Body:**

Success - Goal Reminder Sent Successfully

No Need - Goal not Achieved not yet, No Notification sent

Failed 500 - Failed to send Notification

**2. Send Goal Achievement Notification**

* **Endpoint:** POST /api/notification/send-goal-achievement/
* **Description:** Send a notification to a user when they achieve their health goals.
* **Parameters:**
  + **userId** *(query)* -The ID of the user to whom the achievement notification will be sent (required).
* **Responses:200 OK**
  + - **Response Body:**

Goal Achievement Notification Sent Successfully – 200

No Reminder Needed – 200 (Goal Not Achieved yet)

Failed to send Reminder – 500

## Report API Documentation

**Report Endpoints**

**1. Get Weekly User Report**

* **Endpoint:** GET /api/report/user/weekly
* **Description:** Retrieve the weekly health report for a user.
* **Parameters:**
  + **userId** *(query)*:The ID of the user (required).
* **Responses:200 OK**
  + - **Response Body:**

{

"averageSteps": 0,

"averageCalories": 0,

"weightChange": 0

}

**2. Get Weekly User Report (PDF)**

* **Endpoint:** GET /api/report/user/weekly/pdf
* **Description:** Download the weekly health report for a user in PDF format.
* **Parameters:**
  + **userId** *(query)*: The ID of the user (required).
* **Responses:200 OK**
  + - **Response Body:**

PDF document of the weekly report

Note - Haven’t Get Actual PDF from my Side when testing Need to Improve and Test

**3. Get Monthly User Report**

* **Endpoint:** GET /api/report/user/monthly
* **Description:** Retrieve the monthly health report for a user.
* **Parameters:**
  + **userId** *(query)*: The ID of the user (required).
* **Responses:200 OK**
  + - **Response Body:**

{

"averageSteps": 0,

"averageCalories": 0,

"weightChange": 0

}

**4. Get Monthly User Report (PDF)**

* **Endpoint:** GET /api/report/user/monthly/pdf
* **Description:** Download the monthly health report for a user in PDF format.
* **Parameters:**
  + **userId** *(query)*:The ID of the user (required).
* **Responses:200 OK**
  + - **Response Body:**

PDF document of the monthly report

Note - Haven’t Get Actual PDF from my Side when testing Need to Improve and Test