Created By: Vishal Gupta

Email: vishal0205gupta@gmail.com

Mobile: 7003605011

API Specification Document

- *Pre-Validation of inputs are not done as of now. Let me know if needed.*
- *Let me know if any other api is required.*
- *Let me know if any change in logic for heart beat calculation is required.*
- *Let me know any other change is required*

1. POST /api/heartbeat/report/evaluated/

Description:

The API takes a json payload with both patient and clinical data and outputs with the evaluated heartbeat report, that is provide high and low data per 15 mins slots. If api param '**save**' is provided with value '**true**' then it saves the data in the database else not. If a report for the same patient is provided then the patient table is just updated and the clinical_data table will be inserted with all the reports without any duplicate check. The output evaluates only the current passed heartbeat data.

Method: POST

URL:

`/api/heartbeat/report/evaluated/:save?`

URL Parameters:

Parameter	Type	Description	Required	Example
save	string	Optional flag to indicate if the report should be saved. Set to `true` to save.	No	`true`

Request Body:

```
{
  "clinical_data": {
    "HEART_RATE": {
      "uom": "beats/min",
      "data": [
        { "on_date": "2020-10-06T06:48:17.503000Z",
"measurement": "111" },
        { "on_date": "2020-10-06T06:48:38.065000Z",
"measurement": "66" },
```

```
        { "on_date": "2020-10-06T06:55:36.001000Z",
"measurement": "148" },
        { "on_date": "2020-10-06T06:59:07.841000Z",
"measurement": "142" },
        { "on_date": "2020-10-06T07:01:35.759000Z",
"measurement": "138" },
        { "on_date": "2020-10-06T07:15:06.645000Z",
"measurement": "134" },
        { "on_date": "2020-10-06T07:38:24.969000Z",
"measurement": "162" },
        { "on_date": "2020-10-06T07:38:29.969000Z",
"measurement": "164" },
        { "on_date": "2020-10-06T07:38:33.969000Z",
"measurement": "164" },
        { "on_date": "2020-10-06T07:38:36.969000Z",
"measurement": "165" },
        { "on_date": "2020-10-06T07:38:40.969000Z",
"measurement": "165" },
        { "on_date": "2020-10-06T07:38:45.969000Z",
"measurement": "167" }
      ],
      "name": "Heart Rate"
    },
    "WEIGHT": {
      "uom": "Kg",
      "name": "Weight"
    },
    "BLOOD_GLUCOSE_LEVELS": {
      "uom": "mmol/L",
      "name": "Blood Glucose"
    },
    "HEIGHT": {
      "uom": "cm",
      "name": "Height"
    },
    "BP": {
      "uom": "mmHg",
      "name": "Blood Pressure"
    },
    "STEPS": {
      "uom": "",
```

```
          "data": [
             { "on_date": "2020-10-05T13:00:00.000000Z",
      "measurement": "11031" },
             { "on_date": "2020-10-06T13:00:00.000000Z",
      "measurement": "4667" },
             { "on_date": "2020-10-07T13:00:00.000000Z",
      "measurement": "13030" },
             { "on_date": "2020-10-08T13:00:00.000000Z",
      "measurement": "3048" }
          ],
          "name": "Steps"
        }
      },
      "patient_id": "gk6dhgh-9a60-4980-bb8b-787bf82689d7",
      "from_healthkit_sync": true,
      "orgId": "8gj4djk6s-a5ad-444b-b58c-358dcbd72dc3",
      "timestamp": "2020-10-09T05:36:31.381Z"
    }
```

Request Body Parameters:

Parameter	Type	Description	Required	Example
clinical_data	object	Contains clinical data for different health parameters.	Yes	See detailed structure in the example above.
patient_id	string	The unique identifier of the patient.	Yes	gk6dhgh-9a60-4980-bb8b-787bf82689d7
from_healthkit_sync	boolean	Indicates whether the data was synced from HealthKit.	Yes	true
orgId	string	The unique identifier for the organization.	Yes	8gj4djk6s-a5ad-444b-b58c-358dcbd72dc3
timestamp	string	Timestamp when the data was generated (ISO 8601 format).	Yes	2020-10-09T05:36:31.381Z

Response:

- **Status Code:** 202 Accepted (for successful report evaluation)
- **Response Body Example (for successful response):**

```json
[
  {
    "from_date": "2020-10-06T06:48:17.503Z",
    "to_date": "2020-10-06T07:01:35.759Z",
    "measurement": {
      "low": "66",
      "high": "148"
    }
  },
  {
    "from_date": "2020-10-06T07:15:06.645Z",
    "to_date": "2020-10-06T07:15:06.645Z",
    "measurement": {
      "low": "134",
      "high": "134"
    }
  },
  {
    "from_date": "2020-10-06T07:38:24.969Z",
    "to_date": "2020-10-06T07:38:45.969Z",
    "measurement": {
      "low": "162",
      "high": "167"
    }
  }
]
```

- **Status Code:** `500 Internal Server Error` (in case of an error evaluating the data)
- **Response Body Example (for error response):**

```json
{
  "message": "Error evaluating patient data."
}
```

Response Explanation:

The response will contain a list of **evaluated heartbeat report segments** with the following attributes:

Parameter	Type	Description	Example

from_date	string	The start timestamp of the measurement window.	"2020-10-06T06:48:17.503Z"
to_date	string	The end timestamp of the measurement window.	"2020-10-06T07:01:35.759Z"
measurement	object	Contains the low and high values of the measurement.	{ "low": "66", "high": "148" }

2. GET /api/heartbeat-report/evaluated/

Description:

Provides heart beat data with the required logic, that is provide high and low data per 15 mins slots for all the saved records of the provided patientId.

Method: GET

URL:

/api/heartbeat-report/evaluated/:patientId

URL Parameters:

Parameter	Type	Description	Required	Example
patientId	string	The unique identifier of the patient whose evaluated heartbeat report is being fetched.	Yes	gk6dhgh-9a60-4980-bb8b-787bf82689d7

Response:

- **Status Code: 200 OK (for successful response)**
- **Response Body Example (for successful response):**

```
[
  {
    "from_date": "2020-10-06T06:48:17.503Z",
    "to_date": "2020-10-06T07:01:35.759Z",
    "measurement": {
      "low": "66",
      "high": "148"
    }
  },
  {
    "from_date": "2020-10-06T07:15:06.645Z",
    "to_date": "2020-10-06T07:15:06.645Z",
    "measurement": {
      "low": "134",
      "high": "134"
    }
  },
  {
    "from_date": "2020-10-06T07:38:24.969Z",
    "to_date": "2020-10-06T07:38:45.969Z",
```

```
      "measurement": {
        "low": "162",
        "high": "167"
      }
    }
  ]
```

- **Status Code: 404 Not Found (when no reports are found for the patient)**
- **Response Body Example (for no reports found):**

```
{

  "message": "No reports found for this patient. Patient_id:
gk6dhgh-9a60-4980-bb8b-787bf82689d7"

}
```

- **Status Code: 500 Internal Server Error (for errors during the fetching of reports)**
- **Response Body Example (for error fetching reports):**

```
{

  "message": "Error fetching reports."

}
```

3. GET /api/all-raw-report/

Description:

Fetches all raw reports (without any evaluation) for a provided patient ID.

Method: GET

URL:

`/api/all-raw-report/:patientId`

URL Parameters:

Parameter	Type	Description	Required	Example
patientId	string	The unique identifier of the patient whose reports you want to fetch.	Yes	gk6dhgh-9a60-4980-bb8b-7 87bf82689d7

Response:

- **Status Code**: `200 OK`
- **Response Body Example:**

```
[
    {
        "test_name": "Heart Rate",
        "value": "111",
        "uom": "beats/min",
        "on_date": "2020-10-06T06:48:17.503Z"
    },
    {
        "test_name": "Heart Rate",
        "value": "66",
        "uom": "beats/min",
        "on_date": "2020-10-06T06:48:38.065Z"
    },
    {
        "test_name": "Heart Rate",
        "value": "148",
        "uom": "beats/min",
        "on_date": "2020-10-06T06:55:36.001Z"
    },
    {
        "test_name": "Heart Rate",
        "value": "142",
```

```
        "uom": "beats/min",
        "on_date": "2020-10-06T06:59:07.841Z"
    },
    {
        "test_name": "Heart Rate",
        "value": "138",
        "uom": "beats/min",
        "on_date": "2020-10-06T07:01:35.759Z"
    },
    {
        "test_name": "Heart Rate",
        "value": "134",
        "uom": "beats/min",
        "on_date": "2020-10-06T07:15:06.645Z"
    },
    {
        "test_name": "Heart Rate",
        "value": "162",
        "uom": "beats/min",
        "on_date": "2020-10-06T07:38:24.969Z"
    },
    {
        "test_name": "Heart Rate",
        "value": "164",
        "uom": "beats/min",
        "on_date": "2020-10-06T07:38:29.969Z"
    },
    {
        "test_name": "Heart Rate",
        "value": "164",
        "uom": "beats/min",
        "on_date": "2020-10-06T07:38:33.969Z"
    },
    {
        "test_name": "Heart Rate",
        "value": "165",
        "uom": "beats/min",
        "on_date": "2020-10-06T07:38:36.969Z"
    },
    {
        "test_name": "Heart Rate",
```

```
            "value": "165",
            "uom": "beats/min",
            "on_date": "2020-10-06T07:38:40.969Z"
        },
        {

            "test_name": "Heart Rate",
            "value": "167",
            "uom": "beats/min",
            "on_date": "2020-10-06T07:38:45.969Z"
        },
        {

            "test_name": "Steps",
            "value": "11031",
            "uom": "",
            "on_date": "2020-10-05T13:00:00.000Z"
        },
        {

            "test_name": "Steps",
            "value": "4667",
            "uom": "",
            "on_date": "2020-10-06T13:00:00.000Z"
        },
        {

            "test_name": "Steps",
            "value": "13030",
            "uom": "",
            "on_date": "2020-10-07T13:00:00.000Z"
        },
        {

            "test_name": "Steps",
            "value": "3048",
            "uom": "",
            "on_date": "2020-10-08T13:00:00.000Z"
        }
    ]
```

- **Status Code**: 404 Not Found
- **Response Body Example:**

```
    {
```

```
        "message": "No reports found for this patient. patient_id:
        gk6dhgh-9a60-4980-bb8b-787bf82689d7"
    }
```

- **Status Code**: 500 Internal Server Error
- **Response Body Example:**

```
{
    "message": "Error fetching reports."
}
```

This logic assumes that the server will return the full list of reports when they are found, return a 404 status with an appropriate message if the reports do not exist, or a 500 status if there's an error during the process.

4. GET /api/patient/

Description:

Fetches the details of one specific patient if `:patientId` is provided, or fetches details of all patients if no `:patientId` is provided.

Method: GET

URL:

`/api/patient/:patientId?`

URL Parameters:

Parameter	Type	Description	Required	Example
patientId	string	The unique identifier of the patient to fetch. If not provided, all patients will be fetched.	No	gk6dhgh-9a60-4980-bb8b-787bf82689d7

Response:

- **Status Code**: 200 OK
- **Response Body** (Example for single patient):

```
[{
  "id": 1,
  "patient_id": "gk6dhgh-9a60-4980-bb8b-787bf82689d7",
  "from_healthkit_sync": true,
  "orgId": "8gj4djk6s-a5ad-444b-b58c-358dcbd72dc3",
  "timestamp": "2020-10-09T05:36:31.381Z"
}]
```

- **Response Body** (Example for all patients):

```
[
  {
    "id": 1,
    "patient_id": "gk6dhgh-9a60-4980-bb8b-787bf82689d7",
    "from_healthkit_sync": true,
    "orgId": "8gj4djk6s-a5ad-444b-b58c-358dcbd72dc3",
    "timestamp": "2020-10-09T05:36:31.381Z"
```

```
      },
      {
        "id": 2,
        "patient_id": "abcd1234-5678-9101-1121-31415",
        "from_healthkit_sync": false,
        "orgId": "1a2b3c4d-5678-9101-1121-31415",
        "timestamp": "2020-10-09T06:00:00.000Z"
      }
    ]
```

5. DELETE /api/patient/

Description:

This endpoint deletes a patient from the `patient` table. And due to the foreign key constraint with cascading deletes, all reports in the `clinical_data` table related to the patient will also be deleted.

Method: DELETE

URL:

`/api/patient/:patientId`

URL Parameters:

Parameter	Type	Description	Required	Example
patientId	string	The unique identifier of the patient to be deleted	Yes	gk6dhgh-9a60-4980-bb8b-787bf82689d7

Response:

- **Status Code**: 200 OK
- **Response Body Example**:

  ```
  {
      "message": "Patient gk6dhgh-9a60-4980-bb8b-787bf82689d7
  and all his reports are deleted successfully."
  }
  ```

- **Status Code**: 404 Not Found
- **Response Body Example**:

  ```
  {
      "message": "No patient found with the provided
  patient_id: gk6dhgh-9a60-4980-bb8b-787bf82689d7"
  }
  ```

- **Status Code**: 500 Internal Server Error
- **Response Body Example**:

  ```
  {
      "message": "Error deleting patient."
  }
  ```

6. DELETE /api/report/

Description:

This endpoint deletes all reports associated with the provided patient ID (`clinical_data` table). It does not delete the patient itself (`patient` table).

Method: DELETE

URL:

`/api/report/:patientId`

URL Parameters:

Parameter	Type	Description	Required	Example
patientId	string	The unique identifier of the patient whose reports need to be deleted.	Yes	gk6dhgh-9a60-4980-bb8b-787bf82689d7

Response:

- **Status Code**: 200 OK
- **Response Body Example**:

```
{
    "message": "Patient gk6dhgh-9a60-4980-bb8b-787bf82689d7
reports deleted successfully."
}
```

- **Status Code**: 404 Not Found
- **Response Body Example**:

```
{
    "message": "No report found with the provided
patient_id: gk6dhgh-9a60-4980-bb8b-787bf82689d7"
}
```

- **Status Code**: 500 Internal Server Error
- **Response Body Example**:

```
{
    "message": "Error deleting patient reports."
}
```

7. DELETE /api/clearAll

Description:

THIS IS FOR TESTING ONLY.
This endpoint clears all tables in the database. It will remove all data, including patients, reports, and any related information.

Method: DELETE

URL:

/api/clearAll

Response:

- **Status Code**: 200 OK
- **Response Body**:

```
{
  "message": "All tables cleared."
}
```

- **Status Code**: 500 Internal Server Error
- **Response Body Example**:

```
{
    "message": "Error clearing tables."
}
```