

GPTitor API Design

Alexandra Starikova-Nasibullina

June 18, 2024

1 Create New Conversation

URL: http://10.100.30.244:8000/conversations

Type: External

Method: POST

Parameters: task_id: string(\$uuid)

Response Type: application/json

Example of Response:

Status Code: 201

```
{
  "conversation_id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "user_id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "messages": [],
  "title": "string",
  "keywords": [
    "string"
  ],
  "task_id": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
}
```

Response Headers: content-length, content-type, date, server **Description:** When a user picks a new task, a new conversation is created.

2 Delete Conversation

URL: http://10.100.30.244:8000/conversations

Type: External

Method: DELETE

Parameters: conversation_id: string(\$uuid)

Response Type: application/json

Example of Response:

Status Code: 200

```
{
  "conversation_id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "user_id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "messages": [],
  "title": "string",
  "keywords": [
    "string"
  ],
  "task_id": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
}
```

Status Code: 404

```
{
  "detail": "Conversation not found"
}
```

Response Headers: content-length, content-type, date, server

Description: Deleting a conversation from the user's history.

3 Send Query

URL: http://10.100.30.244:8000/conversations/{conversation_id}/messages

Type: External

Method: PUT

Parameters: conversation_id: string(\$uuid)

Body:

```
{
  "query_text": "Example query"
}
```

Response Type: application/json

Example of Response:

Status Code: 201

```
{
  "user_id": "b6e2c050-01ca-4bba-8d6f-40db0874b7d5",
  "query_id": "37c8c25f-3be4-4717-ae52-7dd32d8a3524",
  "query_text": "Example query",
  "response_id": "0eb32120-d9c0-449a-9d1c-628e4fe0be24",
  "response_text": "Chatbot response to: Example query",
  "query_score": [
    1.3,
    1.4,
    1.5,
    1.7
  ],
  "comment": "Chatbot comment to: Example query",
  "conversation_id": "11111111-1111-1111-1111-111111111111"
}
```

Status Code: 404

```
{
  "detail": "Conversation not found"
}
```

Status Code: 400

```
{
  "detail": "Question cannot be empty"
}
```

Response Headers: content-length, content-type, date, server

Description: When a user sends a query to the service, the response is returned.

4 Get All Messages by Conversation ID

URL: http://10.100.30.244:8000/conversations/{conversation_id}

Type: External

Method: GET

Parameters: conversation_id: string(\$uuid)

Response Type: application/json

Example of Response:

Status Code: 200

```
{
  "conversation_id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "user_id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "messages": [],
  "title": "string",
  "keywords": [
    "string"
  ],
  "task_id": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
}
```

```

}
Status Code: 404
{
  "detail": "Conversation not found"
}

```

Response Headers: content-length, content-type, date, server

Description: Getting all the content of a conversation from the user's history.

5 Get Query-Response Pair by Query ID

URL: http://10.100.30.244:8000/conversations/{conversation_id}/messages/{query_id}

Type: External

Method: GET

Parameters: conversation_id, query_id : string(\$uuid)

Response Type: application/json

Example of Response:

```

Status Code: 200
{
  "user_id": "00000000-0000-0000-0000-000000000000",
  "query_id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "query_text": "Example query",
  "response_id": "00000000-0000-0000-0000-000000000000",
  "response_text": "Example response",
  "query_score": [
    0,
    2.5,
    5,
    10
  ],
  "comment": "Example comment",
  "conversation_id": "00000000-0000-0000-0000-000000000000"
}
Status Code: 404
{
  "detail": "Conversation not found"
}
Status Code: 404
{
  "detail": "Message not found"
}

```

Response Headers: content-length, content-type, date, server

Description: When the user click on the query, the feedback for this query is shown.

6 Get Criterion 1 by Query ID

URL: http://10.100.30.244:8000/feedback/{query_id}/criterion_1

Type: External

Method: GET

Parameters: query_id : string(\$uuid)

Response Type: application/json

Example of Response:

```

Status Code: 200
Response body: 0.5

```

Response Headers: content-length, content-type, date, server

Description: The score for criterion 1 to be shown in the feedback panel and used for metrics.

7 Get Criterion 2 by Query ID

URL: `http://10.100.30.244:8000/feedback/{query_id}/criterion_2`

Type: External

Method: GET

Parameters: `query_id : string($uuid)`

Response Type: `application/json`

Example of Response:

Status Code: 200

Response body: 0.5

Response Headers: `content-length, content-type, date, server`

Description: The score for criterion 2 to be shown in the feedback panel and used for metrics.

8 Get Criterion 3 by Query ID

URL: `http://10.100.30.244:8000/feedback/{query_id}/criterion_3`

Type: External

Method: GET

Parameters: `query_id : string($uuid)`

Response Type: `application/json`

Example of Response:

Status Code: 200

Response body: 0.5

Response Headers: `content-length, content-type, date, server`

Description: The score for criterion 3 to be shown in the feedback panel and used for metrics.

9 Get All Metrics by Query ID

URL: `http://10.100.30.244:8000/feedback/{query_id}/calculate_metrics`

Type: External

Method: GET

Parameters: `query_id : string($uuid)`

Response Type: `application/json`

Example of Response:

Status Code: 200

Response body:

```
[
  1.5,
  2.2
]
```

Response Headers: `content-length, content-type, date, server`

Description: The list of metrics calculated over criteria.