

Chat API Developer Guide

Overview

This is a mental health support chat API built with Django and Ollama. The API provides endpoints for starting chat sessions, sending messages, and receiving AI-powered responses.

Base URL

```
https://controller-incurred-charts-dedicated.trycloudflare.com/api/v1
```

Authentication

Most chat endpoints allow anonymous access (AllowAny permission). Some administrative endpoints require an admin API key for security.

Core Chat Endpoints

1. Health Check

Check if the API and Ollama service are available.

Endpoint: GET /health/

Response:

```
{
  "status": "healthy",
  "ollama_service": "available",
  "model": "command-r7b-arabic:latest",
  "message": "All systems operational"
}
```

cURL Example:

```
curl -X GET "https://controller-incurred-charts-dedicated.trycloudflare.com/api/v1/health/"
```

2. Start Chat Session

Create a new chat session to begin conversation.

Endpoint: POST /chat/start/

Parameters:

- **title (optional):** Custom title for the chat session

Request Body:

```
{
  "title": "My Mental Health Chat"
}
```

Response:

```
{
  "id": "123e4567-e89b-12d3-a456-426614174000",
  "title": "My Mental Health Chat",
  "created_at": "2024-01-01T10:00:00Z",
  "updated_at": "2024-01-01T10:00:00Z",
  "is_active": true,
  "messages": [],
  "message_count": 0
}
```

cURL Example:

```
curl -X POST "https://controller-incurred-charts-dedicated.trycloudflare.com/api/v1/chat/start/" \
-H "Content-Type: application/json" \
-d '{"title": "CLI Chat Session"}'
```

3. Send Message

Send a message to the AI and receive a response.

Endpoint: `POST /chat/message/`

Parameters:

- **session_id (required):** UUID of the chat session
- **message (required):** The message content to send

Request Body:

```
{
  "session_id": "123e4567-e89b-12d3-a456-426614174000",
  "message": "Hello, I'm feeling anxious today"
}
```

Response:

```
{
  "user_message": {
    "id": 1,
    "content": "Hello, I'm feeling anxious today",
    "timestamp": "2024-01-01T10:01:00Z"
  },
  "ai_response": {
    "id": 2,
    "content": "I understand you're feeling anxious today. That's completely normal and I'm here to help. Can you tell me more about",
    "timestamp": "2024-01-01T10:01:05Z"
  }
}
```

cURL Example:

```
curl -X POST "https://controller-incurred-charts-dedicated.trycloudflare.com/api/v1/chat/message/" \
-H "Content-Type: application/json" \
-d '{"session_id":"123e4567-e89b-12d3-a456-426614174000","message":"Hello, I'\''m feeling anxious today"}'
```

Complete Chat Flow Example

Here's a complete example of how to use the chat API:

Step 1: Check Health

```
curl -X GET "https://controller-incurred-charts-dedicated.trycloudflare.com/api/v1/health/"
```

Step 2: Start Chat Session

```
SESSION_RESPONSE=$(curl -s -X POST "https://controller-incurred-charts-dedicated.trycloudflare.com/api/v1/chat/start/" \
-H "Content-Type: application/json" \
-d '{"title": "My Support Session"}')

SESSION_ID=$(echo "$SESSION_RESPONSE" | grep -o '"id": "[^"]*"' | cut -d'"' -f4)
echo "Session ID: $SESSION_ID"
```

Step 3: Send Messages

```
curl -X POST "https://controller-incurred-charts-dedicated.trycloudflare.com/api/v1/chat/message/" \
-H "Content-Type: application/json" \
-d '{"session_id": "$SESSION_ID", "message": "I'm having trouble sleeping lately\","}
```

Step 4: Continue Conversation

```
curl -X POST "https://controller-incurred-charts-dedicated.trycloudflare.com/api/v1/chat/message/" \
-H "Content-Type: application/json" \
-d '{"session_id": "$SESSION_ID", "message": "What can I do to improve my sleep?\","}
```

Error Handling

The API returns appropriate HTTP status codes:

- 200 OK: **Successful request**
- 201 Created: **Resource created successfully**
- 400 Bad Request: **Invalid request data**
- 404 Not Found: **Resource not found**
- 500 Internal Server Error: **Server error**
- 503 Service Unavailable: **Ollama service unavailable**

Error Response Format

```
{
  "error": "Error message description",
  "details": "Additional error details if available"
}
```

Rate Limiting

Currently, no rate limiting is implemented, but it's recommended to implement reasonable delays between requests in production use.

Language Support

The API supports both English and Arabic languages. The AI will respond in the same language as the input message.

Technical Details

AI Model

- **Model:** command-r7b-arabic:latest
- **Service:** Ollama
- **Context:** Maintains conversation history for contextual responses
- **Specialization:** Mental health support with few-shot learning

Database

- **Sessions:** Stored with UUID identifiers

- **Messages:** Linked to sessions with role (user/assistant) and timestamps
- **Anonymous Support:** Sessions can be created without authentication

Security

- CORS enabled for web client integration
- No sensitive data logging
- Optional authentication for enhanced features

This API is designed specifically for mental health support conversations and includes appropriate safeguards and fallback responses for technical issues.