

AIコンシェルジュ「AI API」

AIコンシェルジュ「AI API」 は、リクエスト内容を一切加工することなく AI (LLM/Gemini)へ送信し、生成された回答をそのままレスポンスするAPIです。

※現在は、Gemini Enterprise Agent Platformの「Gemini」での回答となります。

JSプラグインなどと組み合わせることで、案件情報を要約させることや、活動内容をもとに商談結果を採点させるといった、幅広い用途で自由度高くAI活用を行うことが可能になります。

ご利用方法

AIコンシェルジュ「AI API」をご利用いただくためには、以下の3つのステップを実施する必要があります。

- 1. [認証](#)
- 2. [AIへの質問リクエスト](#)
- 3. [AIからの回答取得](#)

1. 認証

APIを利用するためには、まず認証を行いアクセストークンを取得する必要があります。

Request

項目	値
リクエストメソッド	POST
エンドポイント	https://gateway-esm.softbrain.com/auth/token
リクエストヘッダ	Content-Type: application/json
リクエストボディ	下記のJSON形式

```
{
  "grantType": "password",
  "userId": "yourUserId",
  "password": "yourPassword",
  "tenantId": "yourTenantId"
}
```

正常時 Response (200 OK)

```
{
  "accessToken": "${JWT}",
  "tokenType": "BEARER",
  "expireIn": "3600",
  "refreshToken": "${refreshToken}"
}
```

異常時 Response

- HTTP 400 Bad Request: リクエスト内容が正しくない場合に発生します。
- HTTP 500 Internal Server Error: 想定外のエラーの場合に発生します。

2. AIへの質問リクエスト

概要

AIサーバーに対して、質問文を送信します。
案件情報を要約させる指示や、商談結果を採点させる基準など、AI (LLM)に投げかけたい内容を `question` にそのまま記載してご利用ください。
また、`sessionId` を引き継ぐことで、会話の文脈を保持したままAI (LLM)とのマルチターン対話を行うこともできます。

Request

項目	値
リクエストメソッド	POST
エンドポイント	https://gateway-ai.softbrain.com/ai/chat
リクエストヘッダ	• Authorization: Bearer \${accessToken} (「1. 認証」で取得したアクセストークン) • Content-Type: application/json
リクエストボディ	下記のJSON形式

Request Body

フィールド	型	必須	説明
question	string	○	Geminiへそのまま渡す質問文（プロンプト）。加工されずに送信されるため、案件情報の要約指示や採点基準など、Geminiに実行させたい内容をそのまま記載します。
mode	string	○	"NORMAL" を固定で指定します。
sessionId	string	一	会話セッションの識別子。 新規の会話では省略（または null） します。 追加質問（会話の継続）では、直前のレスポンスで返却された <code>sessionId</code> を指定します。

新規質問の例

```
{
  "question": "以下の案件情報を3行で要約してください。 \n---\n{案件情報のテキスト}" ,
  "mode": "NORMAL"
}
```

追加質問（会話継続）の例

前回のレスポンスで得た `sessionId` を指定すると、それまでの会話の文脈を踏まえて回答します。

```
{
  "question": "先ほどの要約をもとに、次にとるべきアクションを1つ提案してください。",
  "mode": "NORMAL",
  "sessionId": "3f2504e0-4f89-41d3-9a0c-0305e82c3301"
}
```

異常時 Response

- HTTP 401 Unauthorized: `Authorization` ヘッダーが無い、またはトークンが無効な場合に発生します。

3. AIからの回答取得

概要

「2. AIへの質問リクエスト」に対する回答は、**Server-Sent Events (SSE)** ストリームとして返却されます。クライアントはストリームを購読し、イベント名ごとに処理を分岐してください。

イベント一覧

イベント名	説明	データ
<code>keepalive</code>	接続維持のために約 25 秒間隔で送信されます。無視して構いません。	空文字
<code>done</code>	回答生成の完了イベント。 AI (LLM)が生成した回答を含みます。このイベントを受信したら処理を終了してください。	JSON（後述）
<code>timeout</code>	サーバー側のタイムアウト（約 5 分）到達時に送信されます。	空文字

`keepalive` を挟みながら、最終的に `done` イベントで回答が届きます。クライアントは `done` を受信した時点でストリームを閉じてください。

`done` イベントのデータ

`done` イベントの `data` には、以下の構造の JSON が含まれます。

フィールド	型	説明
<code>answerData</code>	string	AI (LLM)が生成した回答本文。 加工されないため、Geminiの出力がそのまま返却されます。
<code>sessionId</code>	string	本会話のセッション ID。 追加質問時に <code>sessionId</code> として送信することで会話を継続できます。
<code>mode</code>	string	質問モード。 <code>"NORMAL"</code> が返ります。

`done` イベントの例

```
event: done
data: {"answerData":"1. oo案件は...\n2. 現在の進捗は...\n3. 次に必要なアクションは...", "sessionId":"3f2504e0-4f89-41d3-9a0c-0305e82c3301", "mode":"NORMAL"}
```

異常時 Response

ステータスコード	説明
500 Internal Server Error	サーバー内部エラー（Geminiとの通信エラーなど）。

- Gemini 側で一時的なエラーが発生した場合など、`done` イベントの `answerData` にエラーを示すメッセージが含まれて返却されることがあります。

4. 呼び出し例

cURL

```
curl -N -X POST "https://gateway-ai.softbrain.com/ai/chat" \
-H "Authorization: Bearer <アクセストークン>" \
-H "Content-Type: application/json" \
-d '{
  "question": "以下の案件情報を3行で要約してください。\\n---\\n{案件情報のテキスト}",
  "mode": "NORMAL"
}'
```

`-N`（`--no-buffer`）を付けることで、SSE のイベントを逐次受信できます。

5. 活用例

「AI API」は質問文を一切加工しないため、JS プラグインと組み合わせることで自由度の高い活用ができます。

- **案件情報の要約**：案件の詳細情報をプロンプトに含めて送信し、要点を短くまとめさせる
- **商談結果の採点**：活動内容の履歴をプロンプトに含めて送信し、あらかじめ定めた基準にもとづいて商談結果を採点・評価させる

6. 注意事項

- 1 リクエストあたりのサーバー側タイムアウトは **約 5 分** です。
- 接続維持のための `keepalive` イベントが定期的に届きます。クライアント側でタイムアウトを設定する際は、これを考慮してください。
- 会話の文脈は `sessionId` 単位で管理されます。異なる会話を並行して扱う場合は、それぞれ別の `sessionId` を用いてください。
- リクエスト内容は加工されずに Gemini へそのまま送信されるため、プロンプトに含める案件情報や個人情報の取り扱いにはご注意ください。