

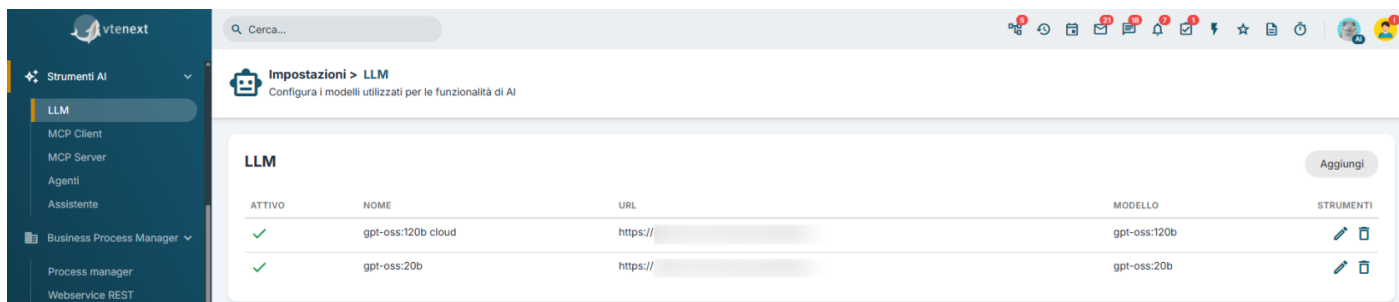
22.1 LLM

La sezione **LLM** delle impostazioni permette di registrare i **modelli linguistici (LLM)** che vtenext può usare per gli agenti e altre funzioni AI collegate. Qui si definiscono i dati di connessione del modello, il nome con cui identificarlo e alcuni parametri che influenzano il comportamento delle risposte.

vtenext non include un modello AI già pronto all'interno del CRM. Per configurare un LLM è necessario disporre di un modello remoto accessibile tramite API compatibili con OpenAI, oppure di un modello locale già installato e raggiungibile. In altre parole, questa configurazione serve a collegare vtenext a un servizio AI esterno o locale: il modello non è "a bordo" del CRM.

La schermata elenco

Nella lista sono visibili le configurazioni già salvate. Per ogni voce puoi vedere:



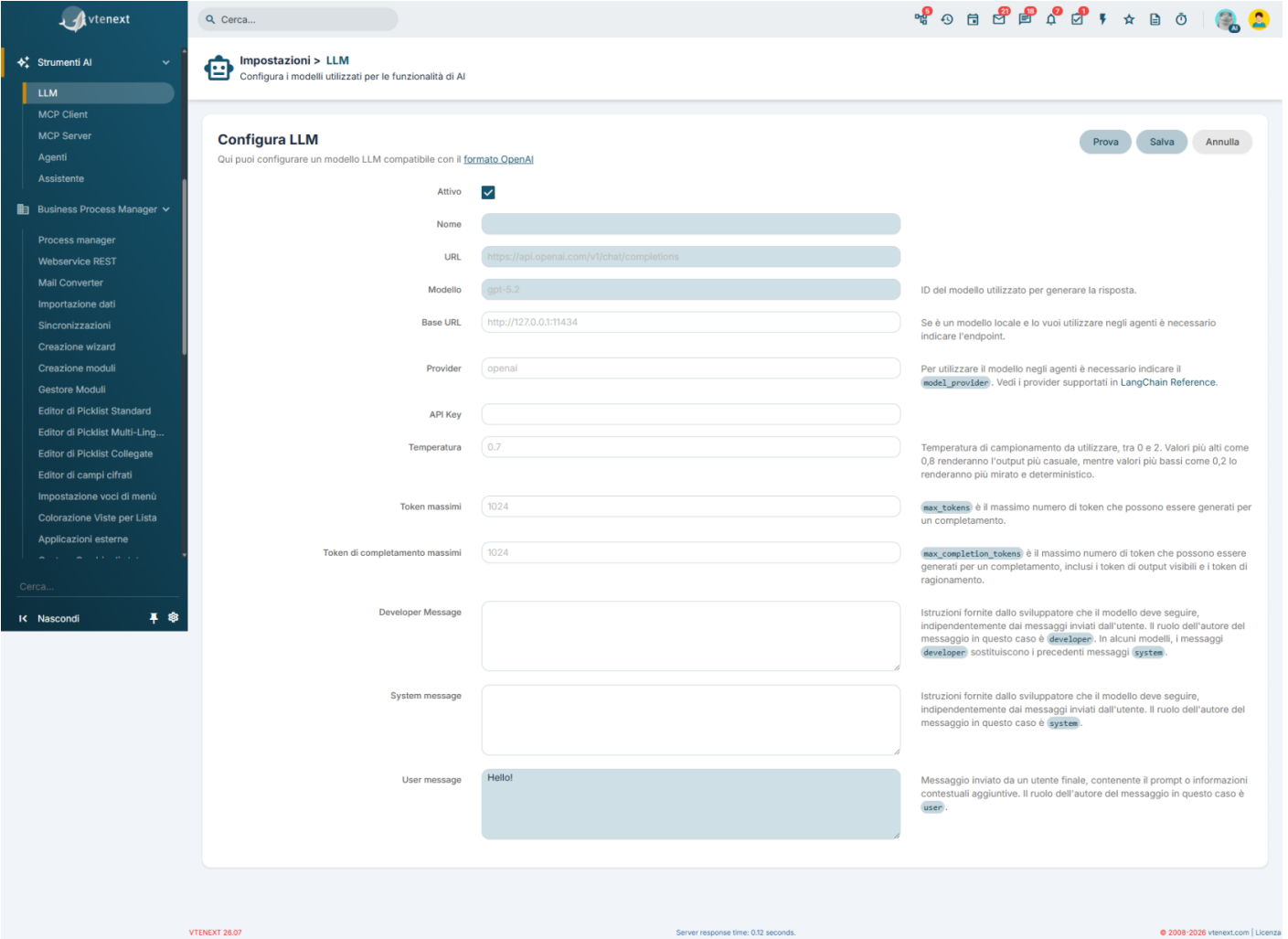
The screenshot shows the vtenext web interface. On the left is a sidebar with a menu containing 'Strumenti AI', 'LLM', 'MCP Client', 'MCP Server', 'Agenti', 'Assistente', 'Business Process Manager', 'Process manager', and 'Webservice REST'. The main area is titled 'Impostazioni > LLM' with a subtitle 'Configura i modelli utilizzati per le funzionalità di AI'. Below this is a table with the heading 'LLM' and an 'Aggiungi' button. The table has five columns: 'ATTIVO', 'NOME', 'URL', 'MODELLO', and 'STRUMENTI'. It contains two rows of data, both with green checkmarks in the 'ATTIVO' column.

ATTIVO	NOME	URL	MODELLO	STRUMENTI
✓	gpt-oss:120b cloud	https://	gpt-oss:120b	 
✓	gpt-oss:20b	https://	gpt-oss:20b	 

- Stato attivo o non attivo
- Nome della configurazione
- URL del servizio
- Modello utilizzato

Dalla lista puoi creare una nuova configurazione, modificare una voce esistente, eliminarla oppure attivarla e disattivarla rapidamente.

Creare un nuovo LLM



1. Apri la sezione LLM.
2. Fai clic su Aggiungi.
3. Compila i campi richiesti.
4. Se necessario, esegui un test della configurazione.
5. Salva.

Base URL	http://127.0.0.1:11434	Se è un modello locale e lo vuoi utilizzare negli agenti è necessario indicare l'endpoint.
Provider	openai	Per utilizzare il modello negli agenti è necessario indicare il <code>model_provider</code> . Vedi i provider supportati in LangChain Reference .
API Key		
Temperatura	0.7	Temperatura di campionamento da utilizzare, tra 0 e 2. Valori più alti come 0,8 renderanno l'output più casuale, mentre valori più bassi come 0,2 lo renderanno più mirato e deterministico.
Token massimi	1024	<code>max_tokens</code> è il massimo numero di token che possono essere generati per un completamento.
Token di completamento massimi	1024	<code>max_completion_tokens</code> è il massimo numero di token che possono essere generati per un completamento, inclusi i token di output visibili e i token di ragionamento.

Attivo	rende il modello disponibile per l'utilizzo
Nome	nome descrittivo del modello visualizzato in vtenext. Campo obbligatorio.

URL	endpoint API a cui vtenext invia le richieste al modello (ad esempio <code>https://api.openai.com/v1/chat/completions</code>). Campo obbligatorio
Modello	identificativo del modello da utilizzare (ad esempio <code>gpt-5.2</code>). Campo obbligatorio.
Base URL	è l'indirizzo di base del servizio che ospita il modello. In pratica dice agli agenti AI e all'orchestratore Python dove si trova il servizio del modello. Diventa particolarmente importante quando si utilizza un modello locale o un servizio interno all'infrastruttura, per esempio un endpoint come <code>http://127.0.0.1:11434</code> .
Provider	indica a vtenext e all'orchestratore Python che tipo di servizio c'è dietro al modello, ad esempio OpenAI oppure Ollama
API Key	chiave di autenticazione fornita dal provider, necessaria per autorizzare le richieste API
Temperatura	controlla il livello di casualità delle risposte generate. Valori bassi (ad esempio <code>0,2</code>) producono risposte più coerenti, prevedibili e ripetibili; valori più alti (ad esempio <code>0,8</code> o superiori) favoriscono risposte più varie e creative
Token massimi	definisce il numero massimo complessivo di token che il modello può utilizzare per elaborare la richiesta, includendo sia i messaggi inviati sia la risposta generata (se supportato dal provider)
Token di completamento massimi	limita il numero massimo di token che il modello può utilizzare esclusivamente per la risposta generata
Developer Message	istruzioni rivolte al modello con ruolo <i>developer</i> , utilizzate per definire regole di comportamento o vincoli applicativi
System Message	istruzioni generali che definiscono il comportamento del modello durante la conversazione
User Message	messaggio di prova inviato al modello per verificarne il funzionamento. Campo obbligatorio

Come funziona il test

Il pulsante di PROVA invia una richiesta reale al modello con i parametri configurati e mostra:

Configura LLM

Qui puoi configurare un modello LLM compatibile con il [formato OpenAI](#)

Attivo ☒

Nome

URL

Modello ID del modello utilizzato per generare la risposta.

Prova

Salva

Annulla

- esito dell'operazione
- codice di risposta
- header restituiti
- corpo completo della risposta

Compatibile con il [formato OpenAI](#)



Risultato della chiamata

RISULTATO

HEADERS

RISPOSTA

Stato **Success**

Codice di ritorno 200 OK

Risultato della chiamata: tab RISULTATO



Risultato della chiamata

RISULTATO

HEADERS

RISPOSTA

```
Content-Type: application/json
Set-Cookie: aid=4dc0d1aa-5357-4285-a355-c9015add5705; Max-Age=31536000; Path=/; Secure; H
x-build-commit: 63edbbb37578481383768a373789def4d0644e45
x-build-time: 2026-07-09T17:42:29-07:00
x-frame-options: DENY
x-request-id: e890cb3b-bf6c-47ee-bace-ee1d81b829d7
Content-Length: 839
Date: Fri, 10 Jul 2026 15:08:14 GMT
Server: Google Frontend
traceparent: 00-1e4ef07affab53e9a548327743036561-c9c9f3982d1fbb43-00
x-cloud-trace-context: 1e4ef07affab53e9a548327743036561/14540420706859989827
Via: 1.1 google
Alt-Svc: h3=":443"; ma=2592000,h3-29=":443"; ma=2592000
Connection: close
```

Risultato della chiamata: tab HEADERS



Risultato della chiamata

RISULTATO

HEADERS

RISPOSTA

Formatta la risposta come: JSON

```
{
  "id": "chatcmpl-363",
  "object": "chat.completion",
  "created": 1783696094,
  "model": "gpt-oss:120b",
  "system_fingerprint": "fp_ollama",
  "choices": [
    {
      "index": 0,
      "message": {
        "role": "assistant",
        "content": "Hello! 🙌 How can I assist you today with VTENEXT? Feel free to let me know what you need—whether it's help with account settings, navigation tips, troubleshooting, or anything else. I'm here to help!",
        "reasoning": "We need to respond as assistant for VTENEXT app. Probably"
```

Risultato della chiamata: tab RISPOSTA

Il test serve a verificare la configurazione,

Il test invia una richiesta reale al modello remoto o locale. Se il modello non è raggiungibile via API, il test non può funzionare.

Esempi pratici

Esempio 1: modello compatibile OpenAI

Usa questa configurazione quando il modello è disponibile come servizio remoto tramite API esterna. Imposta nome, URL, modello e API Key, poi invia un semplice messaggio di test.

Esempio 2: modello locale

Se il modello gira in locale o su infrastruttura interna, compila anche Base URL e Provider. Anche in questo caso il modello deve essere già installato, attivo e raggiungibile via rete.

Revision #14

Created 2026-07-08 13:44:45 UTC by manuel.tagliapietra

Updated 2026-07-13 08:17:10 UTC by Admin