API接口描述

名称	类型	描述	请求参数	响应内容
/agent/invoke	POST	运行智能体并返回大模型结果或中断数据	request=AgentRequest(user_id: str session_id: str query: str system_message: Optional[str])	完成状态: response = AgentResponse(session_id=session_id, status="completed", result=result) 中断状态: response = AgentResponse(session_id=session_id, status="interrupted", interrupt_data=interrupt_data) 失败状态: error_response = AgentResponse(session_id=session_id, status="error", message=f"处理请求时出错: {str(e)}")
/agent/resume	POST	恢复被中断的智能体运行并等待运行完成或再次中断	request=InterruptResponse(user_id: str session_id: str response_type: str args: Optional[Dict[str, Any]] = None)	完成状态: response = AgentResponse(session_id=session_id, status="completed", result=result) 中断状态: response = AgentResponse(session_id=session_id, status="interrupted", interrupt_data=interrupt_data) 失败状态: error_response = AgentResponse(session_id=session_id, status="error", message=f"处理请求时出错: {str(e)}")
/agent/status/{user_id}/{session_id}	GET	获取指定用户当前会话的状态数据	user_id, session_id	response = SessionStatusResponse(user_id=user_id, session_id= session_id, status= status, last_query= last_query, last_updated= last_updated, last_response= last_response)
/agent/active/sessionid/{user_id}	GET	获取指定用户当前最近一次更新的会话ID	user_id	response = ActiveSessionInfoResponse(active_session_id=await app.state.session_manager.get_user_active_session_id(user_id))
/agent/sessionids/{user_id}	GET	获取指定用户的所有会话ID	user_id	response = SessionInfoResponse(active_sessions= active_sessions)
/system/info	GET	获取当前系统内全部的会话状态信息	-	response = SystemInfoResponse(sessions_count= sessions_count, active_users= active_users)

/agent/session/{user_id}/{session_id}	DELETE	删除指定用户当前会话	user_id, session_id	response = {
				"status": "success",
				"message": f"用户 {user_id} 的会话已删除"
				}

/agent/write/longterm	POST	写入指定用户的长期记忆	request=LongMemRequest(
			user_id: str
			memory_info: str
			)
				response = {
				"status": "success",
				"memory_id": result.get("memory_id"),
				"message": result.get("message", "记忆存储成功")
				}

数据模型定义

名称	描述	数据模型
AgentRequest	客户端发起的运行智能体的请求数据	class AgentRequest(BaseModel): # 用户唯一标识 user_id: str # 会话唯一标识 session_id: str # 用户的问题 query: str # 系统提示词 system_message: Optional[str] = "你会使用工具来帮助用户。如果工具使用被拒绝，请提示用户。"
LongMemRequest	客户端发起的写入长期记忆的请求数据	LongMemRequest(BaseModel): # 用户唯一标识 user_id: str # 写入的内容 memory_info: str
AgentResponse	运行智能体后返回的响应数据	class AgentResponse(BaseModel): # 会话唯一标识 session_id: str # 三个状态：interrupted, completed, error status: str # 时间戳 timestamp: float = Field(default_factory=lambda: time.time()) # error时的提示消息 message: Optional[str] = None # completed时的结果消息 result: Optional[Dict[str, Any]] = None # interrupted时的中断消息 interrupt_data: Optional[Dict[str, Any]] = None
InterruptResponse	客户端发起的恢复智能体运行的中断反馈请求数据	class InterruptResponse(BaseModel): # 用户唯一标识 user_id: str # 会话唯一标识 session_id: str # 中断响应类型 response_type: str # 如果是edit, response类型，可能需要额外的参数 args: Optional[Dict[str, Any]] = None
SystemInfoResponse	系统内的会话状态响应数据	class SystemInfoResponse(BaseModel): # 当前系统内会话总数 sessions_count: int # 系统内当前活跃的用户和会话 active_users: List[str]
SessionInfoResponse	所有会话ID响应数据	class SessionInfoResponse(BaseModel): # 当前活跃的用户 active_sessions: List[str]
ActiveSessionInfoResponse	当前最近一次更新的会话ID响应	class ActiveSessionInfoResponse(BaseModel): # 最近一次更新的会话ID active_session_id: str
SessionStatusResponse	会话状态详情响应数据	class SessionStatusResponse(BaseModel): # 用户唯一标识 user_id: str # 会话唯一标识 session_id: Optional[str] = None # 状态：not_found, idle, running, interrupted, completed, error status: str # error时的提示消息 message: Optional[str] = None # 上次查询 last_query: Optional[str] = None # 上次更新时间 last_updated: Optional[float] = None # 上次响应 last_response: Optional[AgentResponse] = None

中断响应类型

名称	描述
accept	接受工具调用
reject	拒绝工具调用
edit	修改工具参数后调用工具
response	不调用工具直接反馈信息

会话状态

名称	描述
not_found	未发现会话
idle	会话空闲
running	会话运行中
interrupted	会话已中断
completed	会话已完成
error	未知会话错误