

核心文件职能详解

我们将项目文件分为数据层、业务组件层、布局层和配置层，下面逐一解析。

省流版本：后端只需要查看 `types.ts` 和 `mockWorkflow.ts` 两个文件即可：

- `types.ts`：定义了前后端交接的数据结构、字段定义
- `mockWorkflow.ts`：后端需要遵循的业务处理流程

1. 数据契约层 (Data Layer)

存放位置： `src/api/`

这一层是前后端协作的基石。

文件名	专业定义	关键职责
<code>types.ts</code>	类型定义文件 (Interfaces & Types)	定义了什么是“任务”，什么是“消息”，什么是“图表数据”。 前后端对接必须严格遵守此文件定义的字段。
<code>mockWorkflow.ts</code>	Mock 服务层 (Mock Service)	在真实后端未就绪时，模拟 Agent 的思考、工具调用延时和数据返回。 调度负责人未来将在此处接入真实 API。

2. 布局与状态容器层 (Layout Layer)

存放位置： `src/layouts/`

这是整个前端应用的“总指挥部”。

文件名	解释	专业定义	关键职责
<code>MainLayout.tsx</code>	“大脑”	布局容器	1. 状态管理 ：维护聊天记录、任务状态、Loading 状态。 2. 组件组装 ：将左侧聊天和右侧面板拼在一起。 3. 逻辑分发 ：接收用户输入，调用 API，并将结果分发给子组件。

3. 业务组件层 (Component Layer)

存放位置： `src/components/`

3.1 聊天模块 (`components/chat/`)

UI文件，不用修改。

文件名	解释	关键职责
<code>ChatPanel1.tsx</code>	“聊天窗口”	渲染消息列表，负责收集用户输入，发送指令。

文件名	解释	关键职责
MessageBubble.tsx	“气泡”	纯展示组件，负责渲染单条消息。

3.2 workflow模块 (components/workflow/) —— 协作重灾区

展示 Agent 的思考过程和结果。

文件名	对应 UI 面板	关键职责
workflowOverview.tsx	任务总览	展示步骤条（意图识别 -> 工具调用 -> 结果生成）。显示绿色的勾或转圈的 Loading。
ToolLogPanel.tsx	工具日志	展示黑底绿字的日志。用于调试工具调用的入参和出参是否正确。
ResultPanel.tsx	执行结果	核心展示区 。根据 Agent 返回的数据，动态渲染折线图、柱状图、表格或总结文本。
ToolStatusPanel.tsx	工具状态	展示当前系统接入了哪些工具（如天气、搜索）。相当于一个“能力清单”。

3.3 公共模块 (components/common/)

文件名	关键职责
PageHeader.tsx	顶部的标题栏、Logo 展示。