



Triple-AI: 12306 智能化需求分析与开发实践

第八组 - Triple-AI
软件工程与项目管理

汇报大纲

- 选题与进度
- 需求和场景分解
- 需求到设计
- 需求到测试
- 挑战与应对
- 未来规划

1. 选题与进度

- **选题:** 12306 网页复现
- **核心:** 模拟核心购票流程，探索 AI 协同开发
- **进度:**
 -  **原型系统上线** (前后端已部署)
 -  **Agent 提示词设计** (Observer, Extracter, Standarder)

2. 需求分解: AI Agent workflow

网站行为 → 形式化需求

- **Observer (观察者):** 扫描功能，预设异常场景 (如登录状态)。
- **Extractor (提取器):** 提取显性、隐性及非功能性需求。
- **Standarder (标准化器):** 将自然语言需求转化为 **Given-When-Then** 格式。

需求提取: 挖掘深层与异常

- **Agent: Observer**
- **深层需求:**
 - 用户说: "我要查票"
 - 真实意图: "帮我找到最合适的出行方案"
- **异常情况:**
 - **未登录** 状态下的功能限制
 - **网络中断** 或 **无效输入**

3. 需求转化设计: TDD workflow

形式化需求 → 可运行代码

- **Designer (设计师):** 理解需求，规划实现路径。
- **Test Generator (测试生成器):** 自动生成单元/集成测试用例。
- **Developer (开发者):** 编写代码以通过所有测试。

4. 需求转化测试: Test Generator

- 输入: `Given-When-Then` 场景描述
- 输出: 自动化测试脚本 (Jest / Vitest)
- 目标: 确保代码实现严格符合业务场景

5. 挑战与应对

- **Agent 自动切换:** 如何实现 Agent 间的无缝调用?
- **AI 测试盲区:** 如何测试购票、支付等真实交互?
- **Issue 自动化:** 如何将需求自动同步到项目管理工具?
- **非功能性需求:** 如何衡量与测试性能、安全性?

我们的解决方案

- **Agent 切换:** 构建一个中央 **Orchestrator (调度器)**。
- **测试盲区: Mocking (模拟)** 外部服务 / **人工确认** 关键节点。
- **Issue 管理:** 调用 **GitHub API** 自动创建和链接 Issues。
- **非功能性需求:** 引入专门的 **性能/安全 Agent**。

6. 未来规划

- **Q4 2025:**
 - [] 完善 12306 需求库
 - [] 构建并集成 GitHub Issue 系统
 - [] 实现 Agent 工作流的初步自动化
 - [] 量化记录开发成果与效率

感谢聆听

Q&A