

# 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

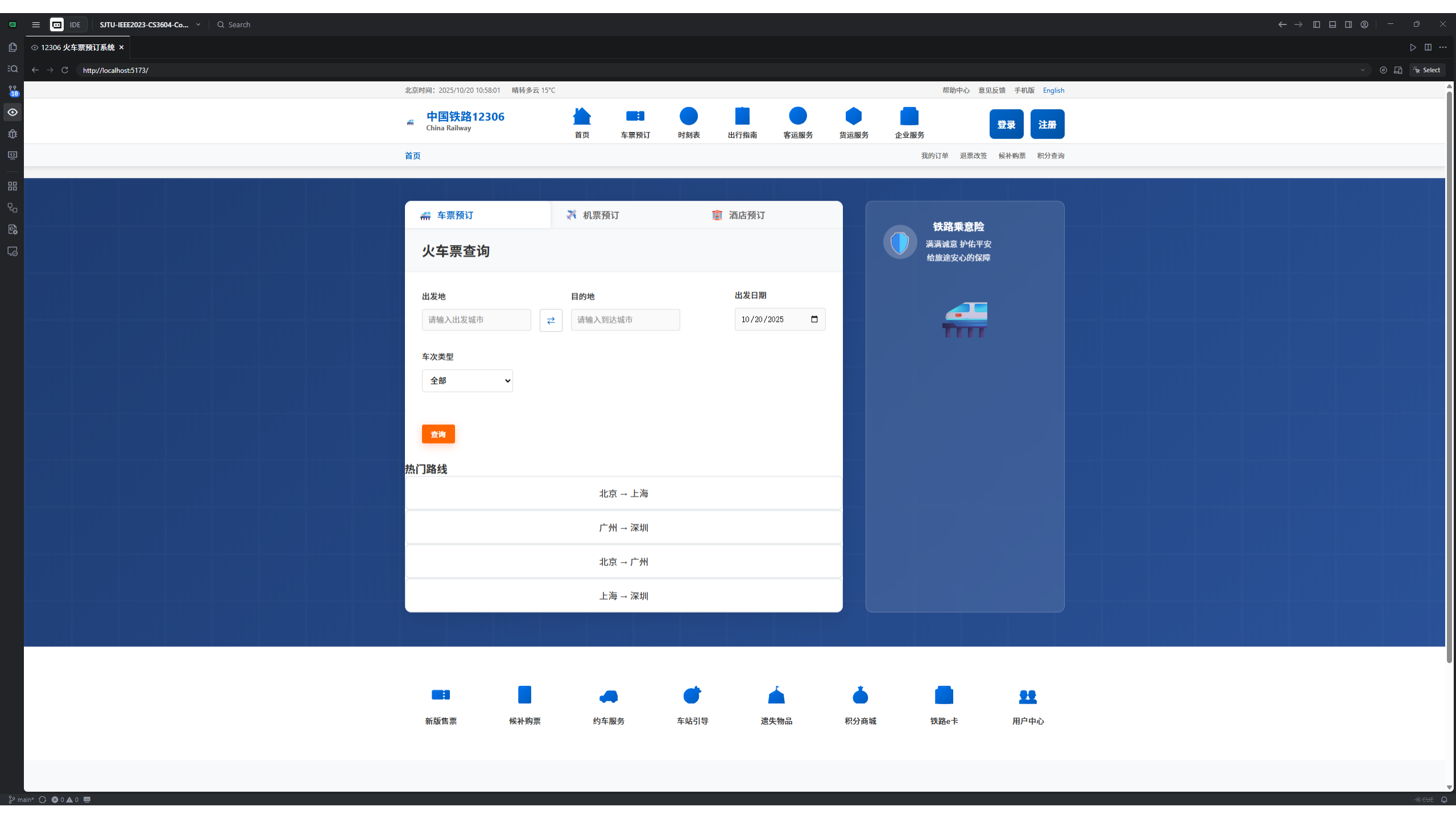


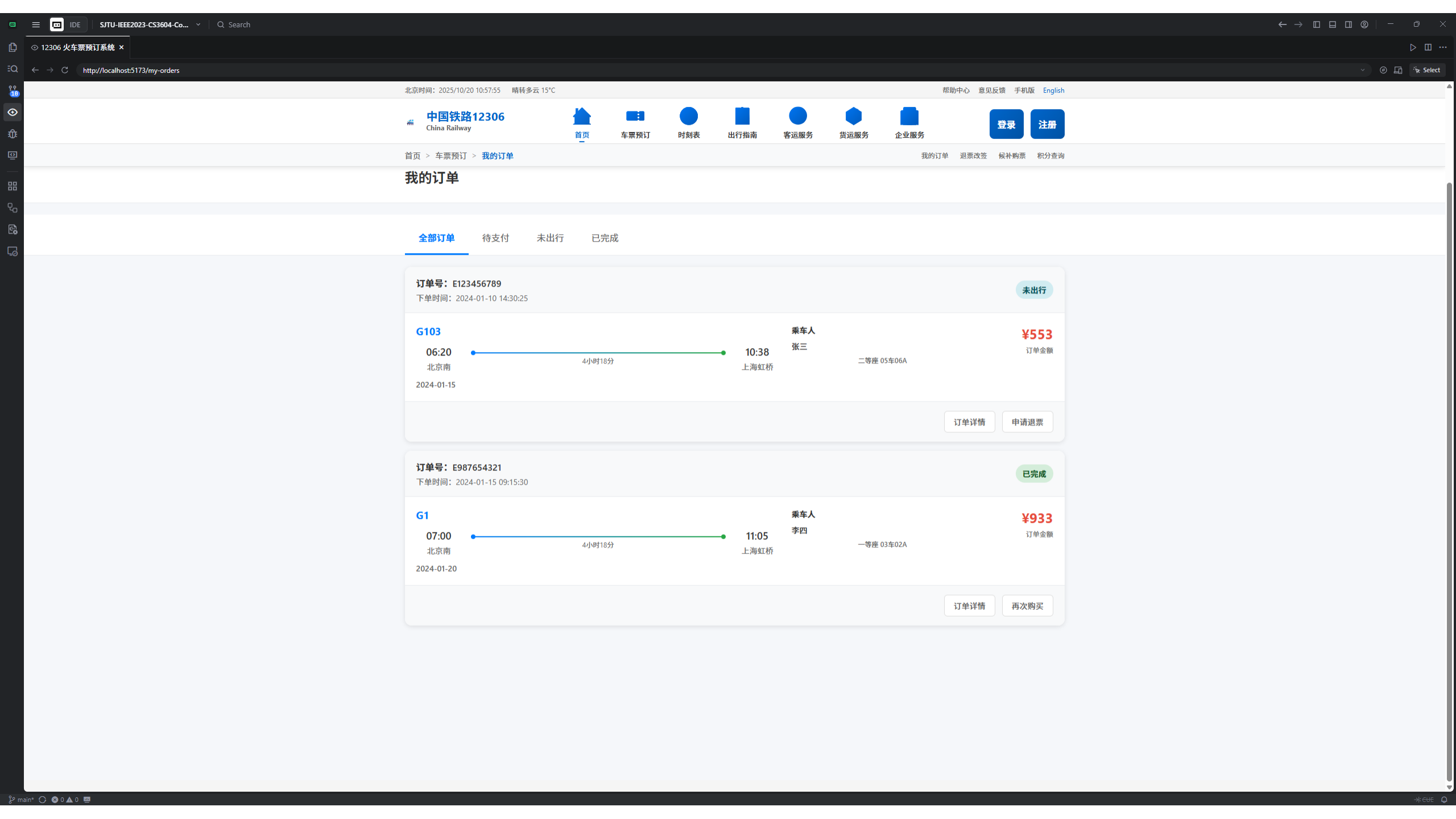
# 汇报大纲

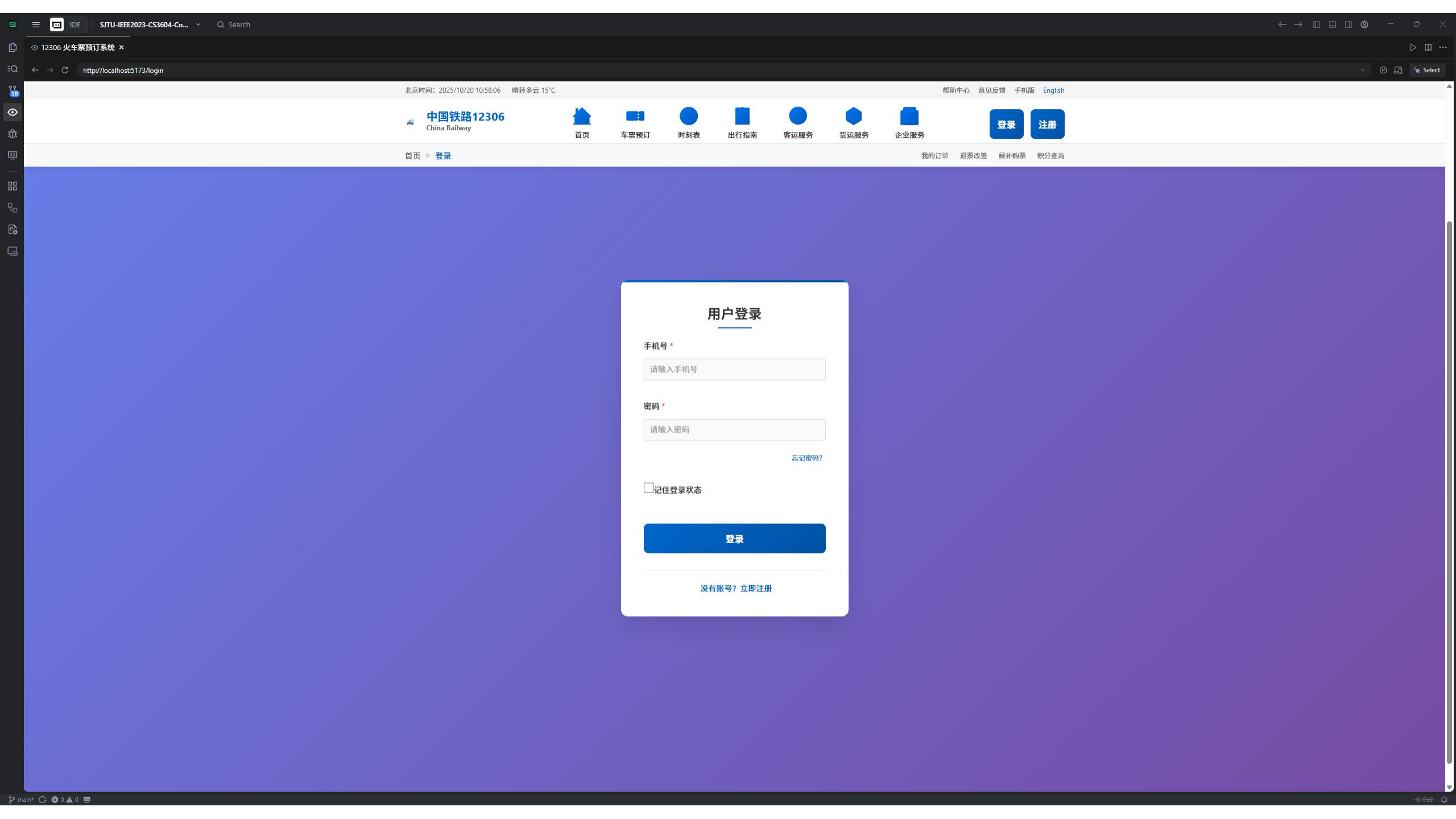
- 选题与进度
- 需求和场景分解
- 需求到设计
- 需求到测试
- 挑战与应对
- 未来规划

# 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, Extractor, Standarder)







IDE

SJTU-IEEE2023-CS3604-Co...

Search

12306 火车票预订系统

http://localhost:5173/login

北京时间: 2025/10/20 10:58:06 晴转多云 15°C

帮助中心 意见反馈 手机版 English

中国铁路12306  
China Railway



首页



车票预订



时刻表



出行指南



客运服务



货运服务



企业服务

登录

注册

首页 > 登录

我的订单 退票改签 候补购票 积分查询

## 用户登录

手机号 \*

请输入手机号

密码 \*

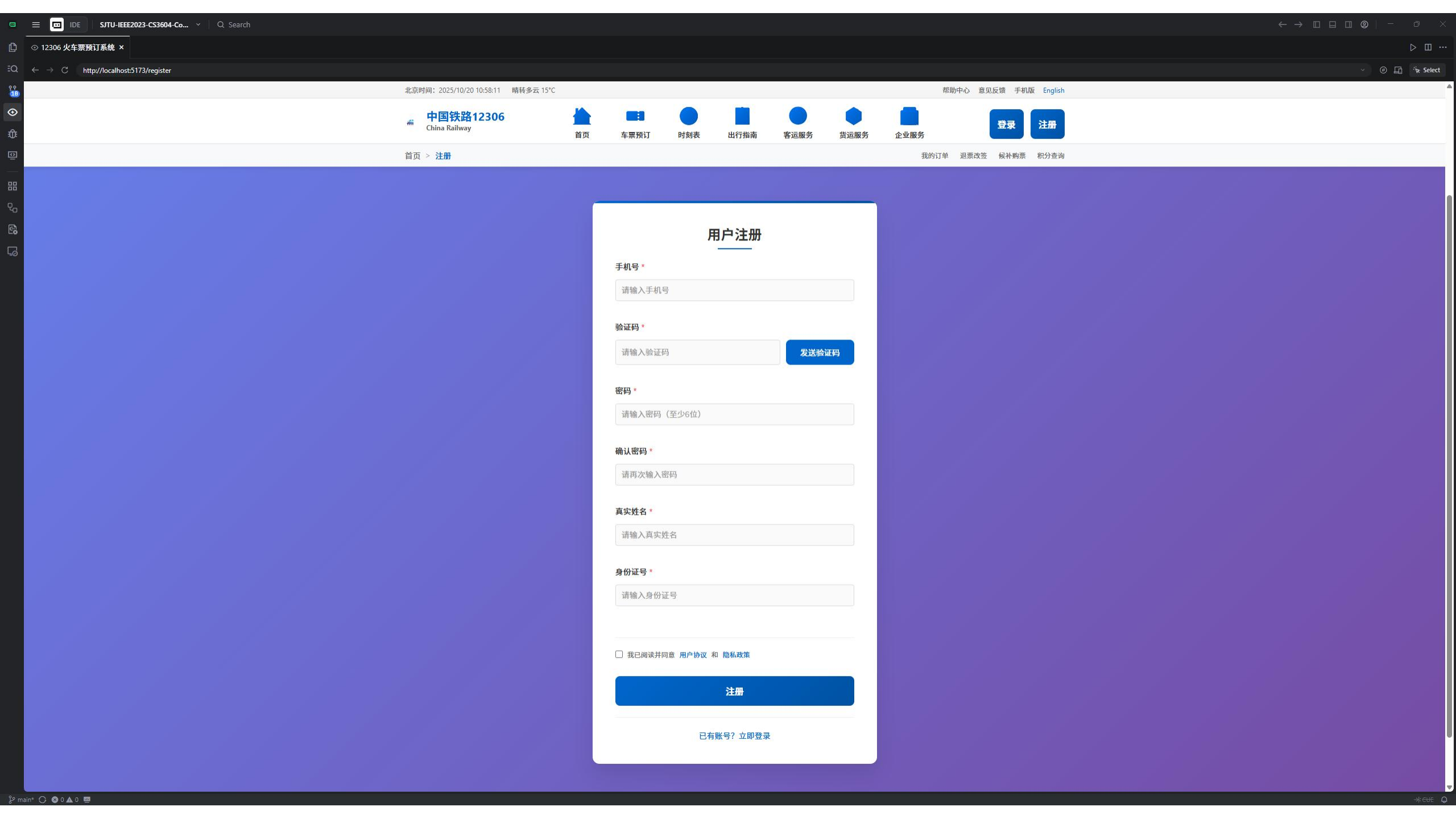
请输入密码

[忘记密码?](#)

☐ 记住登录状态

登录

[没有账号? 立即注册](#)



## 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。



# REQUIREMENT DECOMPOSITION: AI Agent Workflow

Website Behavior > Formalized Requirements



# 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**

# REQUIREMENT EXTRACTION: Uncovering Deep & Anomaly Needs

## EXPLICIT REQUIREMENTS

- UI Featres
- Visible Functions
- User Input Fields



## IMPLIT & NON-FUNCTIONAL REQUIREMENTS

- Performance (Speed, Load Time)
- Security (Data Protection)
- User Experience, (Usability)
- Reliability
- Error Handling



## AI AGENT: Observer

### DEEP NEEDS

→ User says: "Find me tickets"  
True Intent: "Find the best travel options"

### ANOMALY CASES

- Not Logged Loin
- Network Interupation
- Invalid Input

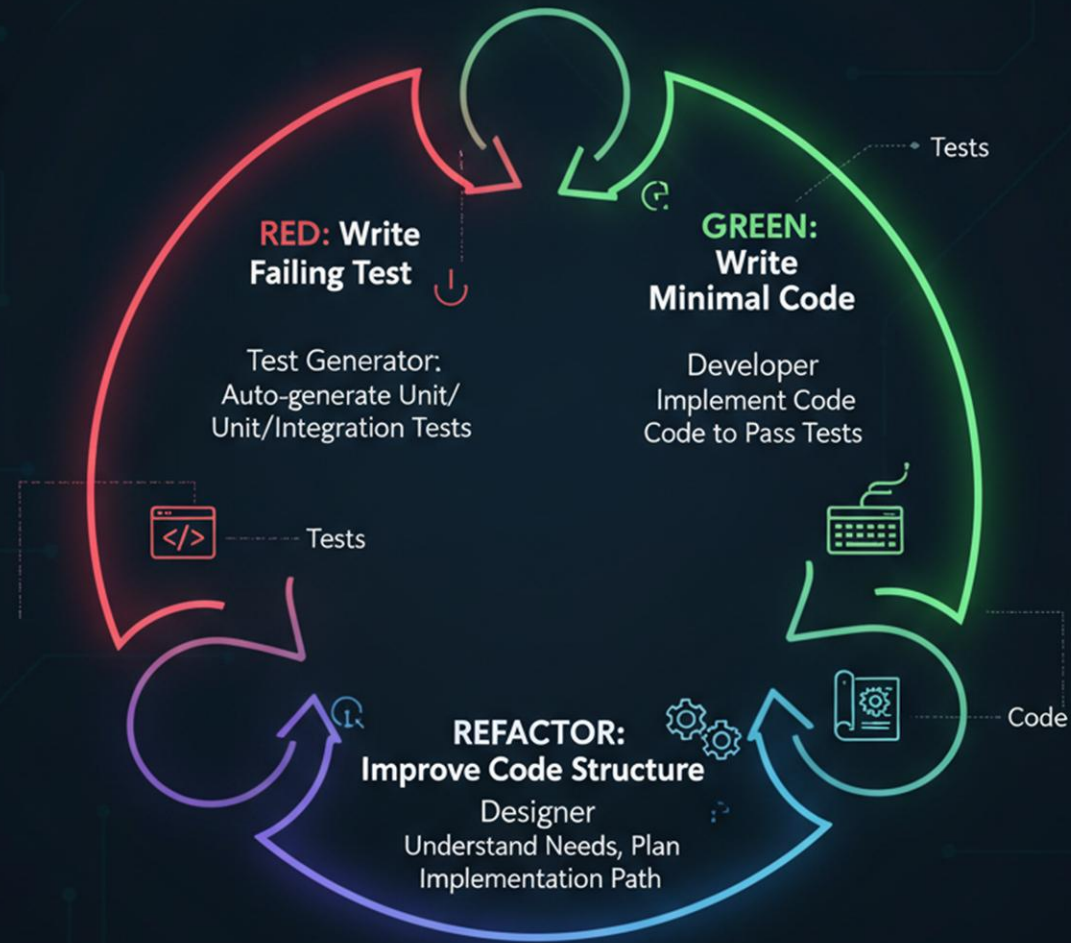
## 3. 需求转化设计: TDD workflow

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。

# REQUIREMENT CONVERSION TO DESIGN: TDD Workflow

Formalized Requirements > Executable Code



## 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?

# OPEN ISSUES



**AGENT SWITCHING:**  
Relay Race?

?



**AI TESTING  
BLINDSPOTS:**  
Real-world Interactions?

?



**ISSUE AUTOMATION**  
Synchronize with PM Tools?

?



**NON-FUNCTIONAL NEEDS:**  
Measure Performance?

?

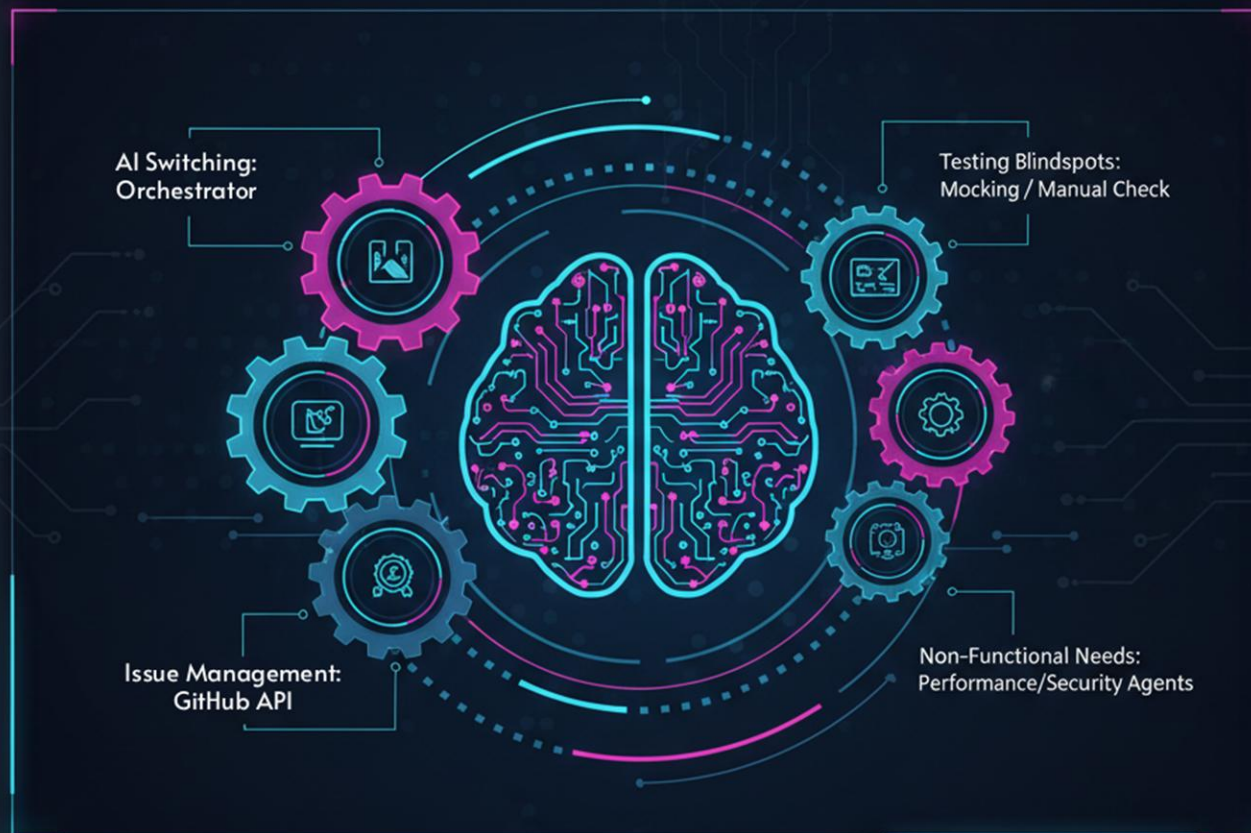


# 我们的解决方案

- **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**。

# OUR SOLUTIONS

Intelligent & Automated Problem-Solving



## 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