# 产品需求文档（PRD）

## 一、产品名称

测试资源中台平台（TestOps Center）

## 二、产品概述

该系统是一个自建型测试资源中台平台，旨在集中管理以下资源： - UI / 接口 / 性能测试用例（支持 YAML/JSON 格式） - 测试脚本（UI自动化、接口压测脚本） - Bug / 缺陷记录（含截图、日志、请求体） - 测试执行记录（Agent 执行的脚本结果） - AI 辅助分析（断言建议、接口分类、报告生成）

系统作为测试 AI 平台的基础支撑层，需具备测试任务调度、测试数据结构化存储、脚本回放与版本管理等能力。

## 三、目标用户

* 测试开发人员
* 自动化测试执行者（本地 Agent）
* 项目经理（查阅用例覆盖与缺陷统计）

## 四、核心功能模块

### 4.1 测试用例管理

* 新建 / 编辑 / 删除测试用例（支持 YAML 与文本编辑器）
* 按项目 / 模块 / 标签分类管理
* 支持版本记录与对比（后期）
* 用例字段：id、name、type（UI / 接口 / 性能）、project、module、tags、created\_by、created\_at、content

### 4.2 测试脚本管理

* 脚本上传（.py / .js / .http / .k6.js）
* 脚本类型识别与分类
* 绑定测试用例（用于回放与执行）
* 支持 YAML 转自动化脚本（AI 分析器接入）

### 4.3 Bug / 缺陷管理

* Bug 提交页面：title、steps、expected、actual、screenshot、related\_case\_id、log
* 状态流转：open → in progress → fixed → closed
* 可选来源：手工创建、Agent 自动上报（失败截图）

### 4.4 测试执行记录

* 显示每个测试用例的执行历史
  + agent\_id、执行时间、status（pass/fail）、断言详情、请求记录、截图路径
* 可下载执行报告（JSON / Markdown / HTML）
* 执行记录与 Bug 建立自动链接（失败自动转为缺陷）

### 4.5 AI 分析模块（预留）

* 用例补全建议
* 请求结构分析 → 自动生成断言
* 自然语言转测试脚本
* DOM + 请求行为分析 → 推荐 Mock 构造 / 压测路径

## 五、系统角色与权限（MVP 简化）

* 管理员：全权限
* 测试人员：仅能编辑自己创建的用例与 Bug
* Agent：只允许拉取任务与上传执行数据（通过 token 校验）

## 六、基础数据模型（摘要）

### 6.1 TestCase

{  
 "id": "tc-001",  
 "name": "登录功能测试",  
 "type": "UI",  
 "project": "电商系统",  
 "module": "登录页",  
 "tags": ["正向", "冒烟"],  
 "content": "YAML 脚本",  
 "created\_by": "grady",  
 "created\_at": "2025-08-07"  
}

### 6.2 Bug

{  
 "id": "bug-001",  
 "title": "登录失败提示异常",  
 "steps": "输入错误密码点击登录",  
 "expected": "提示密码错误",  
 "actual": "无提示",  
 "status": "open",  
 "screenshot": "/uploads/bug-001.png",  
 "related\_case\_id": "tc-001"  
}

### 6.3 ExecutionLog

{  
 "task\_id": "tc-001",  
 "agent\_id": "client-01",  
 "status": "failed",  
 "timestamp": "2025-08-07T14:00:00Z",  
 "requests": [...],  
 "assertions": [...],  
 "screenshots": [...]  
}

## 七、技术架构建议

| 层级 | 技术 | 说明 |
| --- | --- | --- |
| 后端 | FastAPI + SQLite（可换 Postgres） | 快速开发、可轻量部署 |
| 前端 | Vue3 + Element Plus | 管理后台组件丰富 |
| 存储 | 文件系统 / OSS / MongoDB（后期） | 存储截图、脚本内容等 |
| Agent | Python CLI + WebSocket 支持 | 与主系统解耦，按需拉脚本执行 |

## 八、MVP 定义

✅ 包含： - 测试用例管理（YAML 为核心） - Bug 管理（含截图上传） - 脚本上传 + 任务记录 - Agent 接口（上传执行记录）

❌ 不包含： - 权限细化 / 多项目隔离 - AI 断言建议 - Web 控制台全流程（先用简化版）

## 九、验收标准

* 可创建 / 编辑 / 删除用例和 Bug
* Agent 能拉到任务并上传执行结果
* 后台可看到执行记录与报告结构
* 所有数据可通过 REST API 拉取（方便后期平台接入）

## 十、后续扩展计划

* AI 辅助功能引入（断言补全 / 报告分析）
* 权限控制与多租户支持
* 脚本版本控制 / 用例回滚
* Web 控制台美化 + 图表统计
* 持续集成接入（Webhook / CLI）