

微波研发部门研发管理平台

需求规格说明书（SRS）

文档编号: RDP-SRS-2026-001

版本: V1.0

编制日期: 2026年2月20日

密级: 内部公开

目标读者: AI Agent 实现集群

△ **Agent 实现指引**: 本文档是面向 AI Agent 集群的需求规格说明书。每个功能规格包含明确的**输入/输出/验收标准/数据模型/API规范**, Agent应严格按照规格实现, 不得遗漏。对于标注为 **[MUST]** 的条目为强制实现, **[SHOULD]** 为推荐实现, **[MAY]** 为可选实现。

目录

- 引言
- 系统约束与技术规范
- 数据模型规范
- 功能规格: SRS-PORTAL — 门户界面
- 功能规格: SRS-USER — 用户管理

- 6. 功能规格：SRS-PM — 项目管理
- 7. 功能规格：SRS-DEV — 项目开发
- 8. 功能规格：SRS-SHELF — 产品货架
- 9. 功能规格：SRS-TECH — 技术货架
- 10. 功能规格：SRS-KB — 知识库
- 11. 功能规格：SRS-FORUM — 技术论坛
- 12. 功能规格：SRS-IM — 即时通信
- 13. 非功能规格
- 14. API 设计规范
- 15. 验收测试矩阵

1. 引言

1.1 系统标识

项目	内容
系统名称	微波研发部门研发管理平台（R&D Platform, 简称RDP）
系统代号	RDP
版本目标	V1.0（覆盖一至四期全部功能）
部署环境	离线局域网，Docker Compose单节点
用户规模	30-100人

1.2 Agent 实现约定

实现顺序约定：

- 1. **Phase 1** 标记的功能必须在第一批次实现（门户+用户+项目管理基础+IM集成）

2. **Phase 2** 标记的功能在第二批次实现（流程引擎+项目开发+货架）
3. **Phase 3** 标记的功能在第三批次实现（知识库+论坛+搜索）
4. **Phase 4** 标记的功能在第四批次实现（优化+高级特性）

每个功能规格包含： 功能ID | 描述 | 输入 | 输出 | 前置条件 | 数据模型 | API端点 | 验收标准 | 阶段标记

2. 系统约束与技术规范

2.1 技术栈约束（强制）

层次	技术	版本约束	备注
前端	React + TypeScript + Vite	React 18.x, TS 5.x, Vite 5.x	[MUST] 不可替代
UI库	Ant Design 5	5.x	[MUST] 主UI库
后端	Go (Gin framework)	Go 1.22+, Gin 1.9+	[MUST] 核心API
数据库	PostgreSQL	16.x	[MUST]
缓存	Redis	7.x	[MUST]
搜索	MeiliSearch	1.x	[MUST] Phase 3
Git服务	Gitea (独立部署,API集成)	1.22+	[MUST] Phase 2
认证	Casdoor (独立部署)	Latest	[MUST] Phase 1
IM	Mattermost (独立部署)	Latest Team Edition	[MUST] Phase 1
对象存储	MinIO	Latest	[MUST]
容器化	Docker + Docker Compose	24.x	[MUST]

2.2 编码规范约束

规范项	要求
-----	----

语言	[MUST] 代码注释和变量名使用英文；UI文案使用中文（i18n机制）
API风格	[MUST] RESTful API；路径小写+连字符；版本前缀 <code>/api/v1/</code>
错误处理	[MUST] 统一错误响应格式 <code>{"code": int, "message": string, "data": null}</code>
认证方式	[MUST] JWT Bearer Token；Access Token有效期2小时；Refresh Token 7天
分页	[MUST] 统一分页参数 <code>?page=1&page_size=20</code> ；响应含 <code>total, page, page_size</code>
时间格式	[MUST] ISO 8601 (UTC)： <code>2026-02-20T13:00:00Z</code>
ID生成	[MUST] 雪花算法或ULID，不使用自增ID
前端状态	[SHOULD] Zustand 或 React Context；避免Redux
CSS方案	[SHOULD] Tailwind CSS + Ant Design 主题定制
测试覆盖	[SHOULD] 后端核心逻辑单元测试覆盖率 $\geq 60\%$

2.3 项目结构约束

```

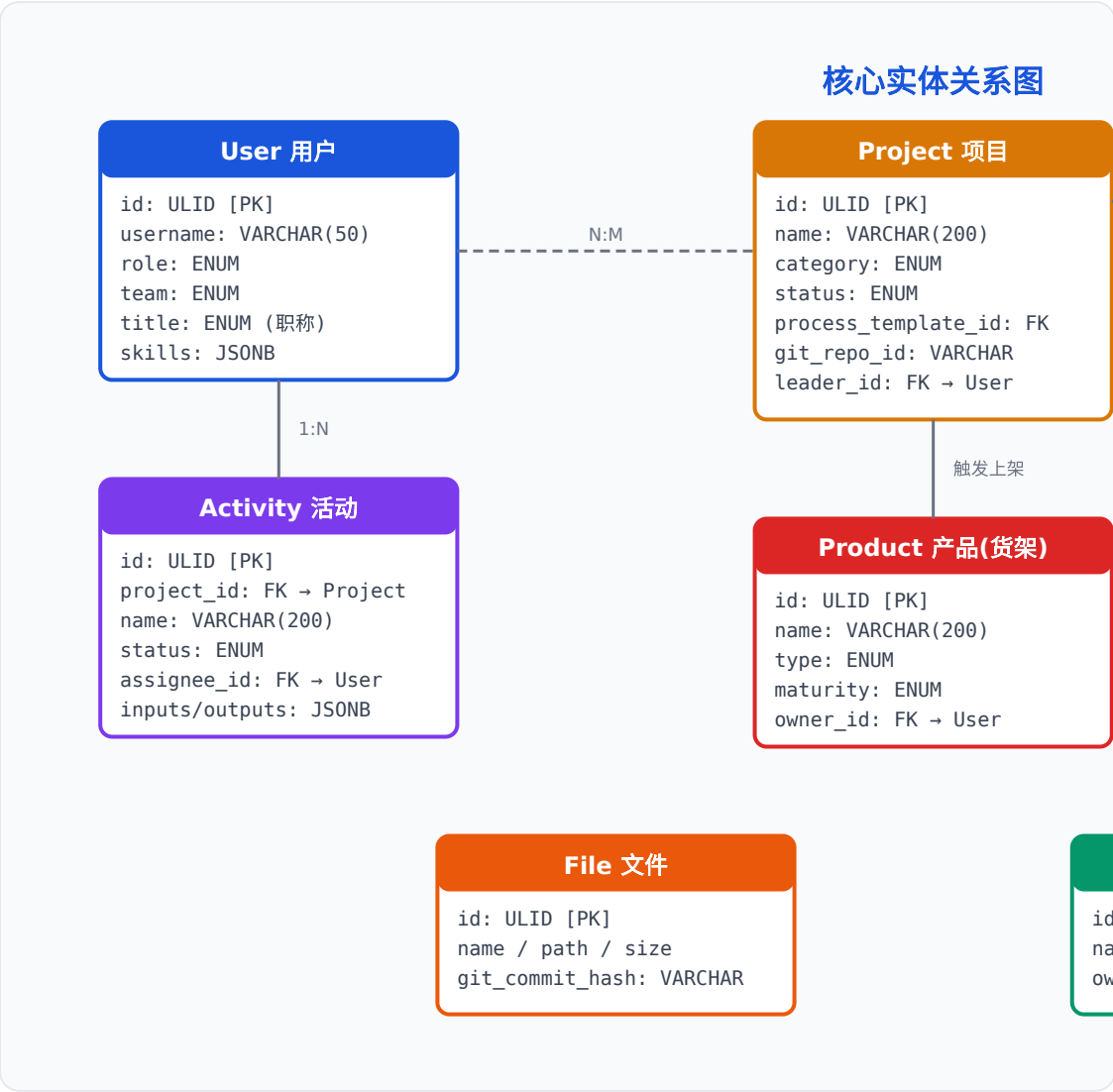
rdp/                                     # Monorepo根目录
├─ apps/
│   ├─ web/                             # 主前端应用(Shell)
│   ├─ desktop/                         # 桌面辅助程序(Electron/Tauri)
│   └─ modules/
│       ├─ user/                        # 用户管理子模块前端
│       ├─ project/                    # 项目管理子模块前端
│       ├─ dev/                        # 项目开发子模块前端
│       ├─ shelf/                      # 产品货架子模块前端
│       ├─ tech-shelf/                 # 技术货架子模块前端
│       ├─ knowledge/                 # 知识库子模块前端
│       └─ forum/                     # 论坛子模块前端
├─ services/
│   ├─ api-gateway/                   # API网关配置(Nginx)
│   ├─ core/                         # 核心业务服务(Go)
│   ├─ workflow/                     # 流程引擎服务(Go)
│   └─ file-service/                 # 文件管理服务(Go)

```

```
|   └─ search-service/      # 搜索索引服务(Go)
|   └─ notification/        # 通知服务(Go)
└─ packages/
|   └─ shared-types/        # 共享TypeScript类型定义
|   └─ shared-utils/        # 共享工具函数
|   └─ ui-components/       # 共享UI组件
└─ database/
|   └─ migrations/          # 数据库迁移脚本
|   └─ seeds/               # 初始数据种子
└─ deploy/
|   └─ docker-compose.yml   # Docker编排
|   └─ nginx/               # Nginx配置
|   └─ scripts/             # 部署脚本
└─ docs/                    # 项目文档
```

3. 数据模型规范

3.1 核心实体关系图（ER图）



3.2 枚举类型定义

```
-- 用户角色
CREATE TYPE user_role AS ENUM ('admin', 'dept_leader', 'team_leader', 'designer',

-- 团队
CREATE TYPE team_type AS ENUM ('product_mgmt', 'product_dev', 'tech_dev', 'general

-- 产品管理子方向
```

```

CREATE TYPE pm_specialty AS ENUM ('model_mgmt', 'reliability');

-- 产品开发产品线
CREATE TYPE pd_product_line AS ENUM ('line_a', 'line_b', 'line_c');

-- 技术开发方向
CREATE TYPE td_specialty AS ENUM ('antenna', 'rf', 'digital', 'power');

-- 设计师职称
CREATE TYPE title_level AS ENUM ('designer_junior', 'assistant_eng', 'engineer', 'senior_engineer');

-- 项目类别
CREATE TYPE project_category AS ENUM (
    'standalone',      -- 单机
    'module',          -- 模块
    'software',         -- 软件
    'tech_dev',        -- 技术开发
    'process_dev',     -- 流程开发
    'knowledge_dev',   -- 知识库开发
    'product_launch'   -- 产品上架
);

-- 项目状态
CREATE TYPE project_status AS ENUM ('draft', 'planning', 'in_progress', 'review', 'complete');

-- 活动状态
CREATE TYPE activity_status AS ENUM ('pending', 'in_progress', 'review', 'complete');

-- 产品成熟度
CREATE TYPE product_maturity AS ENUM ('developing', 'prototype', 'engineering', 'qualified');

-- 产品类型(货架)
CREATE TYPE shelf_type AS ENUM ('standalone', 'module', 'software', 'basic');

-- 基础件子类
CREATE TYPE basic_subtype AS ENUM ('component', 'fastener', 'material');

-- 知识分类
CREATE TYPE knowledge_category AS ENUM (
    'theory', 'standard', 'regulation', 'process', 'case_positive', 'case_negative',
);

-- TRL等级
CREATE TYPE trl_level AS ENUM ('TRL1', 'TRL2', 'TRL3', 'TRL4', 'TRL5', 'TRL6', 'TRL7', 'TRL8');

```


4. SRS-PORTAL — 门户界面模块

SRS-PORTAL-001: 部门门户首页

Phase 1

P0

描述

[MUST] 系统默认首页，未登录用户可见。展示部门简介、新闻公告列表、荣誉展示区、快捷导航入口。

输入

无需认证的公开数据请求

输出

- 部门简介区（可配置的富文本内容）
- 最新公告列表（最近10条，含标题、日期、摘要）
- 荣誉展示轮播（图片+文字）
- 导航卡片入口（各功能模块图标+名称）

API端点

GET /api/v1/portal/announcements?page=1&page_size=10

GET /api/v1/portal/honors

GET /api/v1/portal/config （部门简介等配置信息）

验收标准

- □ 页面在3秒内完成首屏渲染（局域网环境）
- □ 公告列表支持分页加载
- □ 管理员可通过后台编辑门户内容
- □ 响应式适配1920×1080和1366×768分辨率

SRS-PORTAL-002: 个人工作台

Phase 1

P0

描述

[MUST] 用户登录后的默认页面。聚合展示个人相关的待办事项、我的项目列表、消息通知、快捷操作。

组成部分

区域	内容	数据来源
待办事项	待审批项、待完成活动、超期提醒	项目管理+项目开发模块
我的项目	当前参与项目列表，含进度百分比	项目管理模块
消息通知	未读消息数、最近5条通知	通知服务
快捷操作	新建项目、查看货架、进入知识库	固定配置+权限过滤
统计概览	本月完成任务数、贡献度排名	聚合统计

API端点

GET

/api/v1/workbench/todos → 待办列表

GET

/api/v1/workbench/my-projects → 我的项目

GET

/api/v1/workbench/notifications?unread=true → 通知

GET

/api/v1/workbench/stats → 个人统计

验收标准

- 登录后自动跳转到工作台
- 待办事项按紧急程度排序（超期>今日到期>即将到期）
- 点击待办可直接跳转到对应活动/审批页面
- 未读消息实时更新（WebSocket推送）

SRS-PORTAL-003: 全局搜索

Phase 3

P1

描述

[SHOULD] 顶部搜索栏，支持跨模块全文搜索。搜索范围：项目、知识库文档、产品、技术、用户、论坛帖子。

技术实现

[MUST] 基于MeiliSearch实现。各模块数据变更时通过事件驱动更新搜索索引。前端使用防抖（300ms）实时搜索建议。

API端点

```
GET /api/v1/search?q=关键词
&scope=all|project|knowledge|product&page=1
```

验收标准

- □ 搜索响应时间 $\leq 500\text{ms}$ （10万条数据级别）
- □ 支持中文分词和拼音搜索
- □ 搜索结果按相关度排序，关键词高亮
- □ 支持按模块过滤搜索范围

5. SRS-USER — 用户管理模块

SRS-USER-001: 用户CRUD与认证

Phase 1

P0

描述

[MUST] 通过Casdoor实现用户注册、登录、Token管理。支持管理员批量导入(Excel)、单个创建、编辑、停用。首次登录强制修改密码。

数据模型

```
TABLE users (  
  id          ULID PRIMARY KEY,  
  username    VARCHAR(50) UNIQUE NOT NULL,  
  display_name VARCHAR(100) NOT NULL,  
  email       VARCHAR(100),  
  phone       VARCHAR(20),  
  avatar_url  VARCHAR(500),  
  role        user_role NOT NULL DEFAULT 'designer',  
  team        team_type,  
  specialty   VARCHAR(50),      -- 细分方向  
  product_line pd_product_line, -- 产品开发适用  
  title       title_level,  
  skills      JSONB DEFAULT '[]', -- ["微带天线","GaN功放"]  
  honors      JSONB DEFAULT '[]',  
  bio         TEXT,  
  is_active   BOOLEAN DEFAULT true,  
  casdoor_id  VARCHAR(100),     -- Casdoor对应ID  
  created_at  TIMESTAMPTZ DEFAULT NOW(),  
  updated_at  TIMESTAMPTZ DEFAULT NOW()  
);
```

API端点

POST /api/v1/auth/login → {username, password} → {access_token, refresh_token}

POST /api/v1/auth/refresh → {refresh_token} → {access_token}

GET /api/v1/users → 用户列表 (支持?team=&role=&title=筛选)

GET /api/v1/users/:id → 用户详情

POST /api/v1/users → 创建用户 [Admin]

PUT `/api/v1/users/:id` → 更新用户 [Admin]

POST `/api/v1/users/import` → Excel批量导入 [Admin]

PUT `/api/v1/users/:id/role` → 修改角色 [Admin]

验收标准

- ☐ 管理员可创建/编辑/停用用户
- ☐ Excel模板导入≥50个用户无报错
- ☐ 首次登录强制改密后方可进入系统
- ☐ 5次登录失败锁定30分钟
- ☐ JWT Token过期自动刷新，无感知

SRS-USER-002: RBAC权限模型

Phase 1

P0

描述

[MUST] 基于Casbin实现5级角色权限控制。权限继承链：admin > dept_leader > team_leader > designer > other。

权限矩阵（核心）

资源	admin	dept_leader	team_leader	designer	other
用户管理	CRUD	R+部分U	R(本团队)	R(自己)	R(自己)
项目创建	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	申请	<input type="checkbox"/>
项目查看	全部	全部	本团队+参与	参与项目	公开项目
审批	<input type="checkbox"/>	<input type="checkbox"/>	本团队	<input type="checkbox"/>	<input type="checkbox"/>
货架管理	CRUD	CRUD	CRU	R+选用	R
知识库	CRUD	CRUD	CRUD	CR	R

系统配置	□	部分	□	□	□
------	---	----	---	---	---

Casbin模型定义

```
[request_definition]
r = sub, obj, act

[policy_definition]
p = sub, obj, act

[role_definition]
g = _, _

[policy_effect]
e = some(where (p.eft == allow))

[matchers]
m = g(r.sub, p.sub) && keyMatch2(r.obj, p.obj) && r.act == p.act
```

验收标准

- 设计师无法访问用户管理页面（403）
- 团队组长只能看到本团队成员和项目
- 管理员可将部门领导设为管理员角色
- 角色变更后无需重新登录即生效

SRS-USER-003: 个人Profile页面

Phase 1

P1

描述

[SHOULD] 类GitHub个人主页。展示：头像/姓名/职称/团队、技能标签、能力雷达图(ECharts)、贡献热力图(近12月)、项目履历、荣誉列表。

API端点

- GET** `/api/v1/users/:id/profile` → 完整Profile数据
- GET** `/api/v1/users/:id/contributions?year=2026` → 贡献热力图数据
- GET** `/api/v1/users/:id/projects` → 项目履历
- PUT** `/api/v1/users/:id/skills` → 更新技能标签

贡献热力图数据结构

```
// GET /api/v1/users/:id/contributions?year=2026
{
  "data": [
    {"date": "2026-01-01", "count": 3, "details": ["完成活动x2", "提交文档x1"]},
    {"date": "2026-01-02", "count": 0, "details": []},
    ...
  ]
}
```

验收标准

- □ 雷达图展示 ≥ 5 个维度的能力评估
- □ 热力图按日粒度展示全年365天数据
- □ 项目履历按时间倒序，显示角色和状态
- □ 设计师可编辑自己的技能标签和个人简介

6. SRS-PM — 项目管理模块

SRS-PM-001: 项目总览看板Phase 1P0

描述

[MUST] 展示全部门项目的鸟瞰视图。支持按产品线/团队/时间范围/负责人/状态多维筛选和排序。提供列表视图和卡片视图切换。

API端点

GET

/api/v1/projects?category=&status=&team=&leader_id=&date_from=&date_to=&sort=created_at&order=c

验收标准

- 支持≥5个筛选条件组合过滤
- 项目卡片显示：名称、负责人、状态、进度百分比、到期日
- 超期项目自动标红
- 数据按权限过滤（设计师只看到参与项目）

SRS-PM-002: 五步项目创建向导Phase 1P0

描述

[MUST] 引导式项目创建流程：Step1(基本信息) → Step2(类别选择) → Step3(流程绑定) → Step4(团队分配) → Step5(计划确认)。

Step详情

Step	表单字段	约束
1-基本信息	项目名称*, 项目编号*(自动生成可修改), 描述, 计划开始日期*, 计	名称唯一；编号格式: RDP-{CATEGORY}-{YYYYMMDD}-

	划结束日期*	{SEQ}
2-类别选择	项目类别*(单选7种), 产品线(产品开发类必选)	选择后自动推荐匹配的流程模板
3-流程绑定	流程模板*(从推荐列表选择或浏览全部), 预览流程图	流程选定后, 活动列表自动生成
4-团队分配	项目负责人*, 技术负责人, 产品负责人, 质量负责人, 成员列表	从用户列表选择; 可搜索
5-计划确认	活动列表(含工期/负责人), 甘特图预览, 确认创建	可调整各活动工期; 确认后创建项目

创建后自动触发

- [MUST] 创建Gitea仓库（仓库名=项目编号）
- [MUST] 初始化标准目录结构到Git仓库
- [MUST] 创建Mattermost项目频道，添加项目成员
- [MUST] 生成项目活动实例（基于流程模板）
- [SHOULD] 发送项目创建通知给所有成员

API端点

POST /api/v1/projects

Request Body:

```
{
  "name": "X波段TR组件",
  "code": "RDP-STANDALONE-20260220-001",
  "category": "standalone",
  "product_line": "line_a",
  "description": "...",
  "process_template_id": "tpl_xxx",
  "start_date": "2026-03-01",
  "end_date": "2026-12-31",
  "team": {
    "leader_id": "usr_001",
    "tech_leader_id": "usr_002",
    "product_leader_id": "usr_003",
    "members": ["usr_004", "usr_005"]
  },
  "activities": [
    {"template_activity_id": "act_001", "duration_days": 10, "assignee_id": "usr_...",
    ...
  ]
}
```

```
}  
}
```

验收标准

- □ 5步向导可前进/后退/暂存草稿
- □ 创建后自动创建Gitea仓库（通过API验证仓库存在）
- □ 创建后Mattermost频道自动创建（通过API验证）
- □ 流程模板绑定后活动列表正确生成

SRS-PM-003: 甘特图组件

Phase 2

P0

描述

[MUST] 基于gantttask-react实现交互式甘特图。支持任务拖拽调整时间、依赖关系连线（FS/FF/SS/SF）、关键路径高亮、里程碑标记。

功能列表

- [MUST] 按日/周/月三种时间粒度切换
- [MUST] 拖拽调整任务开始/结束时间
- [MUST] 任务依赖关系线（连线+箭头）
- [MUST] 进度条显示（按完成百分比填充）
- [SHOULD] 关键路径自动计算和高亮
- [SHOULD] 基线对比（计划vs实际）
- [MAY] MS Project (.mpp) 导入导出

验收标准

- □ 50个任务+依赖关系的甘特图渲染 ≤ 2秒
- □ 拖拽修改任务时间后，依赖任务自动级联调整
- □ 数据变更实时持久化到后端

SRS-PM-004: 项目文件夹与Git管理

Phase 2

P0

描述

[MUST] 每个项目对应一个Gitea仓库。项目创建时自动初始化标准目录结构。支持通过Web界面浏览文件树、上传/下载文件。支持通过桌面辅助程序将项目文件夹下载至本地，本地修改后同步回服务器。

标准目录结构

```
{项目编号}/
├── 01-需求文档/
├── 02-设计文档/
│   ├── 方案设计/
│   ├── 详细设计/
│   └── 仿真报告/
├── 03-设计文件/
│   ├── 原理图/
│   ├── PCB/
│   └── 结构/
├── 04-测试文档/
├── 05-评审记录/
├── 06-变更记录/
└── README.md
```

API端点

GET	/api/v1/projects/:id/files?path=/	→ 文件树
GET	/api/v1/projects/:id/files/download?path=xxx	→ 下载文件
POST	/api/v1/projects/:id/files/upload	→ 上传文件
GET	/api/v1/projects/:id/files/history?path=xxx	→ 版本历史

验收标准

- 项目创建后Gitea仓库含完整目录结构
- Web端可浏览文件树、查看文件内容（文本类）
- 文件上传自动创建git commit，commit message含文件名
- 版本历史列表显示每次提交的时间、作者、变更说明

7. SRS-DEV — 项目开发模块

SRS-DEV-001: 流程全景视图

Phase 2

P0

描述

[MUST] 以BPMN风格图形化展示项目完整开发流程。当前活动高亮标识（蓝色脉冲），已完成活动（绿色✓），未开始活动（灰色），阻塞活动（红色!）。

交互规格

- [MUST] 点击任意活动节点 → 展开活动详情面板（侧边抽屉）
- [MUST] 详情面板含：活动定义、输入文档列表、输出模板列表、执行指南、参考资料
- [MUST] 进度百分比实时计算： $(\text{已完成活动数} / \text{总活动数}) \times 100$
- [SHOULD] 支持流程图缩放和拖拽平移

验收标准

- ☐ 流程图正确渲染所有活动节点和连线
- ☐ 当前活动有明显的视觉区分（动画高亮）
- ☐ 点击节点 $\leq 300\text{ms}$ 展开详情面板

SRS-DEV-002: 本地软件联动

Phase 2

P0

描述

[MUST] 通过自定义URL协议 `rdp://` 实现Web端点击文件后调用本地软件打开。需配套桌面辅助程序。

协议规范

```
rdp://open?file={local_path}&app={app_name}
```

示例：

```
rdp://open?file=/projects/RDP-001/03-设计文件/原理图/main.schdoc&app=altium
rdp://open?file=/projects/RDP-001/02-设计文档/方案.md&app=obsidian
rdp://open?file=/projects/RDP-001/02-设计文档/报告.docx&app=word
```

桌面辅助程序职责

1. [MUST] 系统安装时注册 `rdp://` 协议处理程序
2. [MUST] 解析协议URL，定位本地文件
3. [MUST] 根据 `app` 参数或文件扩展名调用对应本地软件
4. [MUST] 监听项目文件夹变更（FSWatcher），文件保存后自动执行 `git add + commit + push`
5. [MUST] Commit message自动生成："`完成 {文件名} 编写`"
6. [SHOULD] 系统托盘图标，显示同步状态

文件类型映射表

扩展名	默认应用	备注
.schdoc / .pcbdoc / .PrjPCB	Altium Designer	打开后根据项目名自动创建工程文件
.md	Obsidian 或 Web Markdown编辑器	用户可选
.docx / .xlsx / .pptx	Microsoft Office	—
.pdf	Web内置PDF预览器	不调用本地
.aedt (ANSYS) / .cst	ANSYS / CST Studio	仿真文件

验收标准

- ☐ 点击Web端.schdoc文件，本地Altium Designer正确打开该文件
- ☐ 在Altium中保存文件后，30秒内自动commit到Gitea
- ☐ Web端可看到新的git提交记录
- ☐ 辅助程序无响应时，Web端显示"请安装/启动桌面辅助程序"提示

描述

[MUST] 进入活动后，展示输出物模板列表、执行规范、参考资料（自动关联）。设计师通过Web点击模板开始工作，完成后标记文件完成、活动完成。

活动执行流程

1. 设计师进入活动 → 状态自动变为"进行中" → 更新项目进度
2. 查看输出物模板列表 → 点击模板 → 调用本地软件/Web编辑器创建文件
3. 完成单个文件编辑 → 点击"文件完成" → 自动git commit该文件
4. 所有输出物完成 → 点击"活动完成" → 系统校验完成条件
5. 活动完成 → 自动触发下一活动 → 通知下一活动负责人

自动关联参考资料

[SHOULD] 根据项目类别和当前活动的标签，从知识库中自动匹配相关的参考项目、报告、论文、专利。匹配逻辑：标签交集 ≥ 2 个 的知识条目按相关度排序展示。

API端点

GET	/api/v1/projects/:pid/activities/:aid	→ 活动详情
PUT	/api/v1/projects/:pid/activities/:aid/status	→ {"status": "completed"}
GET	/api/v1/projects/:pid/activities/:aid/templates	→ 输出物模板列表
GET	/api/v1/projects/:pid/activities/:aid/references	→ 自动关联参考资料
POST	/api/v1/projects/:pid/activities/:aid/deliverables	→ 提交交付物
POST	/api/v1/projects/:pid/activities/:aid/feedback	→ 提交反馈/问题

验收标准

- ☐ 活动完成后自动触发下一活动（状态变为pending→可开始）
- ☐ 活动进度变更实时反映在项目管理模块的进度百分比中
- ☐ 参考资料自动关联 ≥ 3 条相关条目（有数据时）
- ☐ 反馈/问题支持@通知和线程回复

8. SRS-SHELF — 产品货架模块

SRS-SHELF-001: 产品浏览与筛选

Phase 2

P0

描述

[MUST] 卡片式产品浏览页面，支持分类浏览（单机/模块/软件/基础）、多维筛选（类型/规格/成熟度/负责人）、关键词搜索。

数据模型

```
TABLE products (  
  id          ULID PRIMARY KEY,  
  name        VARCHAR(200) NOT NULL,  
  code        VARCHAR(50) UNIQUE,  
  type        shelf_type NOT NULL,  
  sub_type    basic_subtype,          -- 基础件适用  
  maturity    product_maturity NOT NULL,  
  description  TEXT,  
  thumbnail_url VARCHAR(500),  
  specs       JSONB,                  -- 技术规格参数  
  tags        TEXT[],  
  owner_id    ULID REFERENCES users(id),  
  source_project_id ULID REFERENCES projects(id), -- 来源项目  
  version     INT DEFAULT 1,  
  parent_id    ULID REFERENCES products(id),    -- 版本链  
  test_data    JSONB,                  -- 测试数据摘要  
  usage_history JSONB DEFAULT '[]',      -- 应用履历  
  issues       JSONB DEFAULT '[]',      -- 问题记录  
  files_repo_id VARCHAR,              -- Gitea仓库ID  
  created_at   TIMESTAMPTZ DEFAULT NOW(),  
  updated_at   TIMESTAMPTZ DEFAULT NOW()  
);
```

API端点

GET /api/v1/products?

type=&maturity=&owner_id=&keyword=&tags=&page=1&page_size=20

GET /api/v1/products/:id → 产品详情（含测试数据、应用履历、问题记录等完整信息）

- POST** `/api/v1/products` → 产品上架
- POST** `/api/v1/products/:id/fork` → 基于产品创建新版本/新项目
- POST** `/api/v1/cart/add` → `{"product_id": "xxx"}` 加入选用清单

验收标准

- 产品卡片展示缩略图、名称、型号、成熟度徽章、负责人头像
- 支持≥4种筛选条件组合
- 产品详情页展示完整的测试数据和应用履历
- "加入选用清单"后可一键将资料包导入项目文件夹

9. SRS-TECH — 技术货架模块

SRS-TECH-001: 技术树与详情

Phase 2

P0

描述

[MUST] 树形/脑图展示部门技术体系。L1: 设计技术/试验技术/仿真技术。L2: 集成/天线/射频/数字/电源/嵌入式等。L3: 具体技术条目。每个技术条目含TRL等级、负责人、关联项目/文件数。

API端点

- GET** `/api/v1/technologies/tree` → 技术树结构
- GET** `/api/v1/technologies/:id` → 技术详情
- POST** `/api/v1/technologies/:id/fork` → fork创建新技术项目

验收标准

- 技术树支持展开/折叠，至少3层深度
- 每个节点显示TRL等级徽章和条目数
- 点击技术条目展示完整详情（描述/文件/关联项目/负责人）

10. SRS-KB — 知识库模块

描述

[MUST] 多级分类体系浏览知识条目。分类树：理论知识(电子书/论文/专利/专业知识) | 标准规范(国标/国军标/航天/行业/部门) | 制度文件 | 流程说明(L1-L4) | 正反面案例 | 仿真模型 | 软件使用指南。

数据模型

```
TABLE knowledge_items (  
  id          ULID PRIMARY KEY,  
  title       VARCHAR(300) NOT NULL,  
  category    knowledge_category NOT NULL,  
  sub_category VARCHAR(100),      -- 二级分类  
  tags        TEXT[] DEFAULT '{}',  
  content     TEXT,              -- Markdown内容(如有)  
  file_path   VARCHAR(500),      -- 文件存储路径  
  file_type   VARCHAR(20),       -- pdf/md/docx/aedt/cst等  
  obsidian_path VARCHAR(500),    -- Obsidian Vault对应路径  
  zotero_key  VARCHAR(100),      -- Zotero条目Key  
  author_id   ULID REFERENCES users(id),  
  view_count  INT DEFAULT 0,  
  download_count INT DEFAULT 0,  
  status      VARCHAR(20) DEFAULT 'published',  
  created_at  TIMESTAMPTZ DEFAULT NOW(),  
  updated_at  TIMESTAMPTZ DEFAULT NOW()  
);  
  
CREATE INDEX idx_kb_tags ON knowledge_items USING GIN(tags);  
CREATE INDEX idx_kb_category ON knowledge_items(category, sub_category);
```

验收标准

- 分类树支持至少三级导航
- 条目支持按标签、分类、关键词筛选
- Markdown文件可Web端渲染预览（含公式、表格、代码高亮）

- ☐ PDF文件可Web端在线预览

SRS-KB-002: Obsidian/Zotero集成

Phase 3

P0

描述

[MUST] 知识库文件路径与Obsidian Vault保持一致。Web端可渲染Obsidian Markdown（含Wiki链接 `[[[]]]`、嵌入 `![[[]]]`、Callout等语法）。Zotero文献库的分类和条目可在Web端浏览，PDF可在线预览。

Obsidian集成规格

- [MUST] Web端知识库根目录 = Obsidian Vault 根目录（路径映射配置）
- [MUST] 支持渲染：`[[WikiLink]]`、`![[嵌入]]`、`> [!NOTE]` Callout、YAML Front Matter
- [MUST] Web端编辑后，文件保存到Vault对应路径
- [SHOULD] 点击"在Obsidian中打开"调用 `obsidian://open?vault=xxx&file=xxx`

Zotero集成规格

- [MUST] 读取Zotero本地数据库或WebDAV同步的文献数据
- [MUST] 展示Zotero分类结构和条目列表
- [SHOULD] 论文PDF在Web端在线预览

验收标准

- ☐ Obsidian Wiki链接在Web端可正确跳转
- ☐ Obsidian Callout语法正确渲染为对应样式
- ☐ Zotero文献列表正确展示分类和条目
- ☐ 编辑保存的文件在Obsidian中可见（路径一致）

11. SRS-FORUM — 技术论坛

描述

[SHOULD] 按技术领域分版块的技术交流论坛。支持富文本/Markdown发帖、回复、@提及、标签分类、全文搜索。

板块结构

天线技术 | 射频技术 | 数字技术 | 电源技术 | 仿真技术 | 通用讨论 | 求助问答

API端点

GET	/api/v1/forum/boards	→ 版块列表
GET	/api/v1/forum/posts?board=&tag=&keyword=&page=1	→ 帖子列表
POST	/api/v1/forum/posts	→ 发帖
POST	/api/v1/forum/posts/:id/replies	→ 回复
PUT	/api/v1/forum/posts/:id/best-reply	→ 标记最佳回复

验收标准

- 发帖支持Markdown格式+图片上传+代码块
- @提及自动发送站内通知
- 最佳答案标记后置顶显示

12. SRS-IM — 即时通信

描述

[MUST] 部署Mattermost Team Edition，通过Nginx反向代理集成到平台。SSO与Casdoor对接，实现统一登录。项目创建时自动创建Mattermost频道。

集成方式

- [MUST] Mattermost通过 `/chat/*` 路径代理，iframe嵌入或独立Tab打开
- [MUST] OAuth2/OpenID Connect对接Casdoor，实现SSO
- [MUST] 通过Mattermost API自动创建项目频道
- [SHOULD] 配置Incoming Webhook接收项目状态变更通知

自动化集成

```
// 项目创建后自动执行：  
1. POST /api/v4/channels → 创建频道 (name=project-{code})  
2. POST /api/v4/channels/{id}/members → 添加项目成员  
3. POST /api/v4/posts → 发送欢迎消息 "项目{name}已创建，欢迎团队成员！"  
  
// 项目状态变更时：  
POST /hooks/{webhook_id} → "项目{name}状态更新为{status}"
```

验收标准

- ☐ 平台登录后无需再次登录Mattermost（SSO生效）
- ☐ 项目创建后Mattermost自动出现对应频道
- ☐ 频道成员与项目团队成员一致
- ☐ 项目状态变更消息自动推送到频道

13. 非功能规格

类别	指标	要求	级别
性能	首屏加载	≤ 3秒（局域网）	[MUST]
	API响应	常规 ≤ 500ms，复杂查询 ≤ 2s	[MUST]
	并发用户	≥ 50人同时在线	[MUST]
	文件上传	单文件 ≤ 500MB，支持断点续传	[SHOULD]
安全	认证	JWT + 可选OTP二次验证	[MUST]
	传输加密	TLS 1.2+（HTTPS）	[MUST]
	密码存储	bcrypt (cost ≥ 10)	[MUST]
	审计日志	登录/权限变更/文件操作全量记录	[MUST]
可用性	系统可用性	工作时间 ≥ 99%	[MUST]
	数据备份	每日自动备份，保留30天	[MUST]
	故障恢复	RTO ≤ 1小时	[SHOULD]
可维护性	日志	结构化JSON日志，支持级别过滤	[MUST]
	健康检查	各服务 /health 端点	[MUST]
	配置管理	环境变量 + .env 文件，支持热重载	[SHOULD]

14. API 设计规范

14.1 统一响应格式

```
// 成功响应
{
```

```

"code": 0,
"message": "success",
"data": { ... },
"meta": {                                // 分页时必须
    "total": 100,
    "page": 1,
    "page_size": 20
}
}

// 错误响应
{
    "code": 40001,                        // 业务错误码
    "message": "用户名已存在",
    "data": null
}

// 错误码规范:
// 400xx - 请求参数错误
// 401xx - 认证错误
// 403xx - 权限不足
// 404xx - 资源不存在
// 500xx - 服务器内部错误

```

14.2 认证头格式

```

Authorization: Bearer {access_token}

// Token刷新:
POST /api/v1/auth/refresh
Body: { "refresh_token": "xxx" }
Response: { "access_token": "new_token", "expires_in": 7200 }

```

15. 验收测试矩阵

测试ID	测试项	测试方法	通过标准	阶段
AT-001	用户登录	输入正确用户名密码	成功跳转工作台, JWT有效	P1
AT-002	权限隔离	设计师访问管理页面	返回403, 页面显示无权限	P1
AT-003	项目创建	5步向导完整走通	项目创建 + Gitea仓库 + MM频道	P1
AT-004	甘特图交互	拖拽调整任务时间	依赖任务级联更新, 数据持久化	P2
AT-005	本地软件调用	点击.schdoc文件	Altium Designer打开对应文件	P2
AT-006	Git自动提交	本地保存文件	30s内Gitea出现新commit	P2
AT-007	活动流转	完成活动点击确认	下一活动激活, 通知发送	P2
AT-008	产品选用	加入购物车+导入项目	产品文件包出现在项目文件夹	P2
AT-009	全文搜索	搜索关键词	≤500ms返回结果, 中文分词正确	P3
AT-010	Obsidian渲染	查看含WikiLink的MD	链接可点击跳转, Callout正确渲染	P3
AT-011	Mattermost SSO	平台登录后访问Chat	无需再次登录	P1
AT-012	并发测试	50用户同时操作	API响应 ≤ 2s, 无500错误	P4

AT-013	备份恢复	模拟数据库故障恢复	1小时内恢复，数据无丢失	P4
AT-014	贡献热力图	查看设计师Profile	365天数据正确渲染	P1
AT-015	项目文件同步	上传文件到服务器	本地和服务器文件一致	P2

Agent 实现总结：

- 本SRS共定义 **9大模块、20+功能规格、15个验收测试用例**
- 核心数据模型包含 **10个实体表、12种枚举类型**
- API端点共计 **40+个RESTful端点**
- 集成的开源服务：Casdoor、Casbin、Gitea、Mattermost、MeiliSearch、MinIO 共**6个**
- 实现时请严格按Phase 1→2→3→4的顺序推进，每个Phase结束时运行对应的验收测试