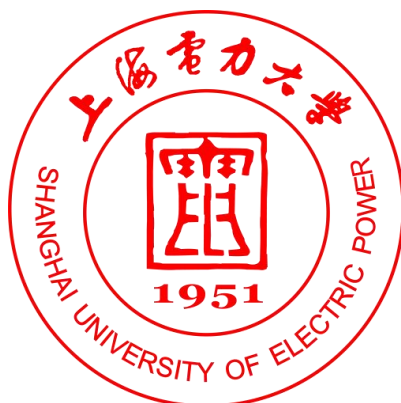


小组作业报告



课程名称 管理信息系统分析与设计

课程代码 2713149.01

专业年级 信管 23

项目名称 OtterSync 项目管理系统

总评分表:

| 学号 | 姓名 | a 考勤 20% (百分制) | 综合评价 80%(百分制) | | | | | 总分 $=a*20\%+f*80\%$ |
|----------|-----|-------------------------|----------------------|----------------------|----------------------|----------------------|----------------------|------------------------|
| | | | b 个人 实验 15% | c 小组 报告 50% | d 小组 答辩 15% | e 个人 贡献 20% | f 汇总分 $f=b+c+d+e$ | |
| 20232310 | 张怡博 | | | | | | | |
| 20232308 | 张运鹏 | | | | | | | |
| 20232285 | 王行健 | | | | | | | |
| 20232309 | 张凯恒 | | | | | | | |

系统分析与设计报告得分表

| 内容 | 考核点（考核细则） | 分值 | 得分 |
|--------------|---------------------------------|------|----|
| 撰写规范性（10） | 格式规范、结构合理、绘图准确（10） | 0-10 | |
| 系统需求分析（40） | 项目背景（2） | 0-2 | |
| | 项目系统可行性分析（4） | 0-4 | |
| | 组织结构及业务流程分析（组织结构图、业务流程图及描述）（10） | 0-10 | |
| | 数据流程分析（分层数据流图及描述）（10） | 0-10 | |
| | 数据汇总分析（E-R 图及关系表、数据字典）（8） | 0-8 | |
| | 功能分析及划分（系统功能结构图及描述）（6） | 0-6 | |
| UML 建模设计（50） | 用例图及用例描述（20） | 0-20 | |
| | 活动图、状态图及相关图的描述（5） | 0-5 | |
| | 类图、类的关系及相关图的描述（20） | 0-20 | |
| | 时序图、协作图及相关图的描述（5） | 0-5 | |

| 学号 | 姓名 | 个人工作及贡献汇总 | 得分 |
|----------|-----|---|----|
| 20232310 | 张怡博 | 1. 参与制作业务流程图、数据流图、状态活动图、用例图、类图、时序图 2. 进行类图描述、活动状态图描述、时序图报告描述汇总 | |
| 20232308 | 张运鹏 | 1. 参与制作业务流程图、数据流图，进行组织机构、业务流程图描述、数据图描述的报告汇总 2. 进行数据汇总分析报告汇总 3. 撰写 Ottersync 系统可行性分析 | |
| 20232285 | 王行健 | 1. 撰写系统背景和研究意义 2. 撰写 Ottersync 系统可行性分析 3. 参与制作用例图，进行用例图描述的报告汇总 | |
| 20232309 | 张凯恒 | 1. 进行对系统功能划分的报告汇总 2. 制作 PPT | |

OtterSync 项目管理系统分析与设计报告

目录

- 目录 3
- 1. 系统背景和研究意义 1
 - 1.1. 系统背景 1
- 2. OtterSync 系统的可行性分析（王行健） 1
 - 2.1. 技术可行性 1
 - 2.2. 经济可行性 (成本/收益) 2
 - 2.3. 操作可行性 2
 - 2.4. 开发思路可行性 2
 - 2.5. 法律可行性 3
- 3. 组织机构及业务流程分析(张运鹏) 3
 - 3.1. 组织结构分析 3
 - 3.2. 业务流程分析 3
 - 3.2.1. 项目与任务管理业务流程图 3
 - 3.2.2. 工作区与群组管理业务流程图 4
 - 3.2.3. AI 智能辅助业务流程图 5
 - 3.2.4. 项目 DashBoard 业务流程图 6
- 4. 数据流程分析(张运鹏) 6
 - 4.1. 分层数据流图 (DFD) 6
 - 4.1.1. 顶层数据流图 7
 - 4.1.2. 第一层数据流图 8
 - 4.1.3. 第二层数据流图 9
- 5. 数据汇总分析(张运鹏) 11
 - 5.1. E-R 图设计 (概念模型) 11
 - 5.2. 关系表 12

| | |
|--------------------------------------|----|
| 5.3. 数据字典 | 12 |
| 5.3.1. 数据项定义 | 13 |
| 5.3.2. 数据结构定义 | 15 |
| 5.3.3. 数据流定义 | 16 |
| 5.3.4. 数据存储 | 18 |
| 5.3.5 处理过程 | 20 |
| 5.3.6 外部实体 | 21 |
| 6. 系统功能及其划分（张凯恒） | 22 |
| 6.1. 系统功能结构图 | 22 |
| 7. 用例图与用例描述（王行健） | 23 |
| 7.1. 用例图 | 23 |
| 7.1.1 开发人员 | 23 |
| 7.1.2 用户 | 24 |
| 7.1.3 项目负责人 | 24 |
| 7.2 用例分析 | 25 |
| 7.2.1 开发者完成开发任务 | 25 |
| 7.2.2 智能问答与检索 | 25 |
| 7.2.3 绩效查看与进度分析 | 26 |
| 7.2.4 总结报告生成 | 27 |
| 8. 活动图、状态图及相关图的描述（张怡博） | 28 |
| 8.1. 任务处理活动图 | 28 |
| 8.2. 任务生命周期状态图 (State Diagram) | 30 |
| 9. 类图、类的关系及相关描述 | 31 |
| 9.1 项目管理系统类图分析 | 31 |
| 9.2 类图图表描述分析 | 32 |
| 10. 时序图、协作图及相关图的描述（张怡博） | 36 |
| 10.1. 时序图 | 36 |

1. 系统背景和研究意义（王行健）

1.1. 系统背景

如今数字化办公逐渐渗透到我们工作流程的方方面面，项目管理软件已经成为团队高效协作、任务有序推进的核心工具。飞书、Jira 这类主流平台凭借完善的任务分配、进度追踪、文档协作功能，在各类企业和团队中得到广泛应用，但这类软件的移动端版本往往存在操作流程繁琐、功能适配性不足的问题，难以满足用户随时随地高效管理项目的需求。

随着人工智能技术的快速迭代，AI 与办公软件的融合成为新的发展方向。为了弥补传统项目管理移动端软件的短板，我们决定开发一款基于 Flutter 和 FastAPI 框架的安卓项目管理软件，既保留飞书、Jira 等传统工具的核心项目管理功能，又创新性地加入 AI 交互模块。用户只需在 Chat 页面发送对话指令，就能让 AI 自动完成新建日程、创建任务、分配负责人等操作，以此简化操作流程，满足开发者提升工作效率、提升项目管理的智能化的需求。

2. OtterSync 系统的可行性分析（王行健、张运鹏）

2.1. 技术可行性

本项目选用 Flutter 作为安卓开发框架，拥有一次编码、多端部署的特性，可从移动端便捷迁移到桌面端。Flutter 通过 Dart 语言编译为原生代码，在安卓端能保证较好的性能，同时其桌面端（Windows、macOS）适配功能已趋于成熟，迁移过程无需大量重写代码，仅需针对桌面端交互逻辑进行少量适配，符合项目同步管理的需求。后端拟采用主流的 Next.js 框架，可实现用户管理、项目数据存储、进度追踪等核心业务逻辑。AI 功能方面，拟集成第三方 api 接口，通过调用 API 实现自然语言识别，将用户在聊天页面输入的自然语言转化为指令对软件进行操作，该方案技术成熟，开发难度可控，符合现有技术能力范围。

2.2. 经济可行性 (成本/收益)

成本主要有三个来源，一是开发成本：无人工成本，所有开发内容由我们自己实现；软件工具均使用免费以及开源版本，无软件购置费用；硬件为当前自有设备，无额外购置成本；第三方 AI 接口前期使用免费额度，后期若需扩展，按最低付费套餐估算。二是测试成本：主要为测试用例设计与执行的时间成本，无额外资金投入；若需进行少量线上测试，可借助免费的测试平台，无测试设备购置费用。三是运营成本：前期无需服务器部署，可使用本地服务器进行测试，后期若需上线供少量用户使用，选用阿里云轻量应用服务器，最低配置每年费用约 300-500 元；域名注册费用每年约 50 元，无其他大额运营成本。

对于收益，若产品功能完善，可面向高校学生社团、小型创业团队等群体提供免费基础版+付费进阶版的服务，进阶版可增加 AI 深度分析、多团队协作等功能，按每人每月 5-10 元的收费标准，可实现少量盈利。

2.3. 操作可行性

本项目目标用户主要为高校学生社团、小型创业团队、个人开发者等，该群体具备一定的互联网使用基础，对 Jira、飞书等项目管理工具的核心功能有基本认知，学习成本低；界面设计将遵循“简洁直观、易于操作”的原则，参考主流项目管理软件布局逻辑，分为项目列表页、任务详情页、AI 聊天页、个人中心页四大核心模块，方便用户快速定位功能。通过短时间的学习即可掌握基本操作，整体操作可行性较高。

2.4. 开发思路可行性

本项目拟采用符合学生开发节奏的迭代式开发思路，分需求分析与原型设计（1-2 周）、核心功能开发（4-6 周）、适配与 AI 集成（3-4 周）、测试与优化（2-3 周）四个阶段推进，将大任务拆解为小模块，各阶段目标与产出明确，便于分工协作与进度把控；功能模块划分遵循模块化与先核心后拓展原则，涵盖用户管理、项目管理、数据同步、AI 交互、通知五大独立模块，可分工实现且便于后期维护，能有效降低开发难度、提升效率；开发团队由 4 名具备相关课程基础的学生组成，分别负责前端、后端、AI 接口集成与测试工作，通过 Git 进行代码版本控制，借助腾讯文档、飞书同步沟通，整体开发思路科学合理，具

备可行性。

2.5. 法律可行性

项目开发与运营将严格遵守《中华人民共和国著作权法》《计算机软件保护条例》等相关法律法规，确保软件代码为自主开发，不抄袭、不盗用他人知识产权。若使用第三方开源代码或组件，将严格遵循开源协议（MIT 协议），注明版权信息，不违反开源条款。

3. 组织机构及业务流程分析(张运鹏、张怡博)

3.1. 组织结构分析

OtterSync 项目管理系统主要服务于软件研发组织或企业的 IT 部门。为了明确系统的用户角色权限及业务流转逻辑，我们对典型的软件研发中心组织结构进行了分析。

该组织最高管理层为研发总监/CTO，下设四个主要职能部门，分别为：产品部、研发部、质量保障部（测试部）与运维部。各部门在 OtterSync 系统中承担不同的职责与权限。

3.2. 业务流程分析

3.2.1. 项目与任务管理业务流程图

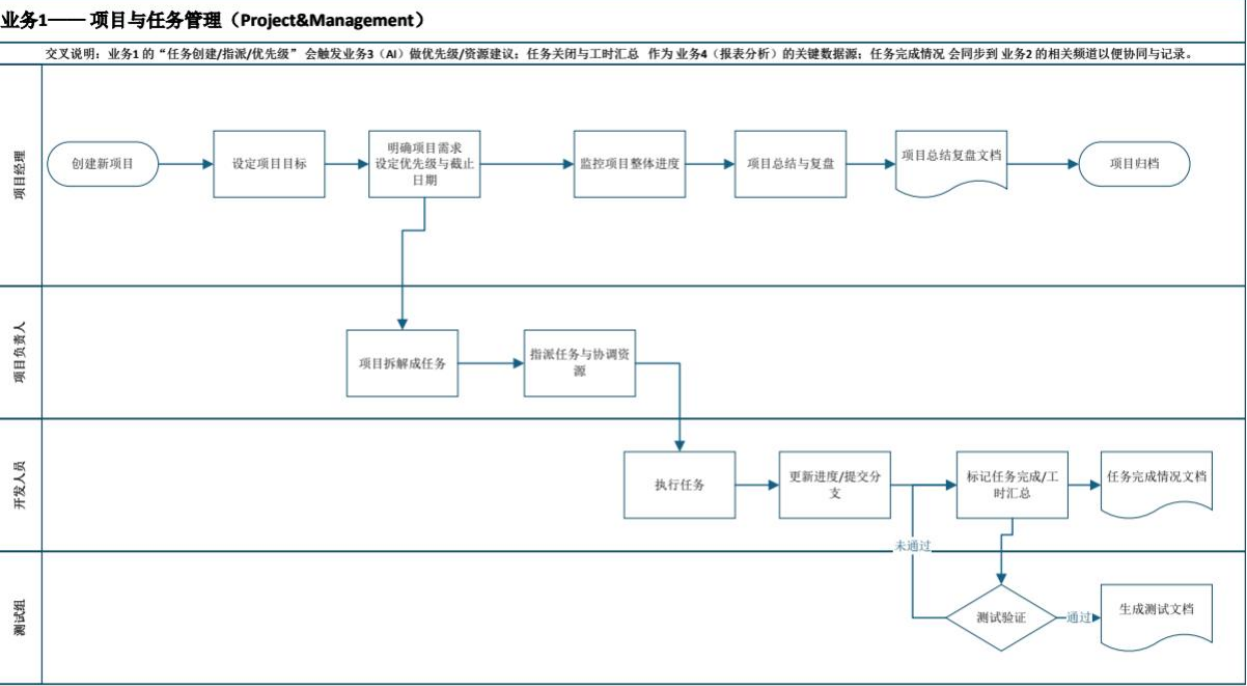


图 1 业务流程图——项目与任务管理

项目与任务管理业务流程：

- 1. 项目创建：项目经理（PM）登录系统，创建新的项目空间，配置项目的基本信息（如名称、代号、成员）。
- 2. 需求录入与规划：产品经理在需求池（Backlog）中录入用户故事（User Story），并在 Sprint 计划会议上将高优先级需求分配至当前的迭代（Sprint）。
- 3. 任务分配与执行：项目经理将任务指派给开发人员。开发人员领取任务，开始开发工作，并将任务状态更新为“进行中”（In Progress）。
- 4. 代码提交与关联：开发人员完成代码编写后，提交代码并关联任务 ID。
- 5. 测试与验收：任务完成后流转至测试人员。测试人员进行验证，若通过则关闭任务（Done），若不通过则打回重做。流程结束。

3.2.2. 工作区与群组管理业务流程图

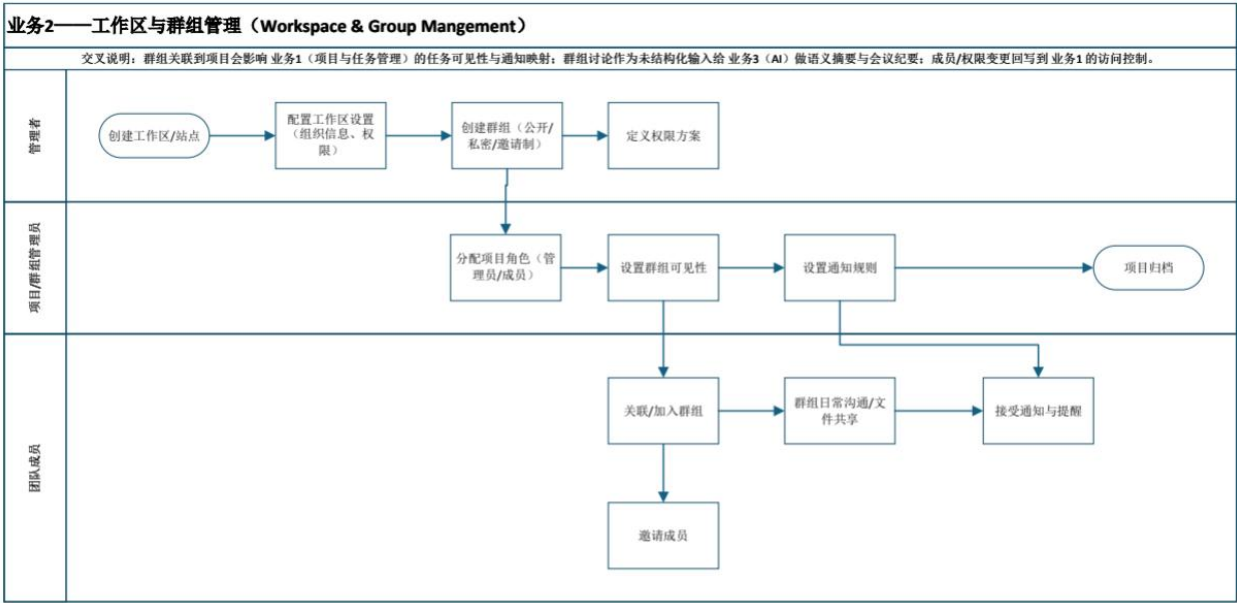


图 2 业务流程图——工作区与群组管理

工作区与群组管理业务流程：

- 1. 工作区初始化：系统管理员创建企业级工作区（Workspace），设定工作区的安全策略和域名。
- 2. 用户邀请与分组：管理员通过邮件邀请团队成员加入工作区，并根据职能部门创建不同的用户组（如“后端组”、“前端组”）。
- 3. 权限分配：管理员为不同的用户组分配相应的系统权限（如“项目管理员”、“普通成员”、“访客”）。
- 4. 成员维护：当有人员入职或离职时，管理员及时添加或移除成员账号，更新群组关系。流程结束。

3.2.3. AI 智能辅助业务流程图

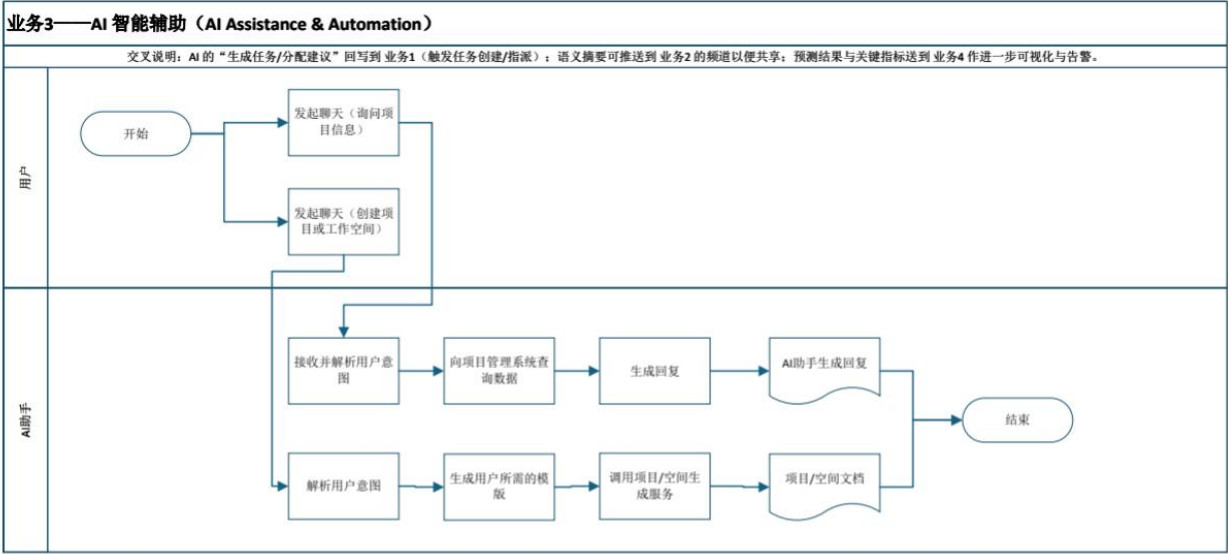


图 3 业务流程图——AI 智能辅助

AI 智能辅助业务流程:

1. 请求输入: 用户在任务详情页或聊天窗口输入自然语言指令 (如“帮我生成这个任务的测试用例”或“总结当前 Sprint 的进度风险”)。
2. 意图识别与分析: AI 模块接收请求, 分析用户的意图和上下文信息 (如当前任务描述、历史评论)。
3. 内容生成: AI 根据分析结果调用大模型接口, 生成相应的文本内容 (如测试步骤、代码建议、风险报告)。
4. 结果反馈与确认: 系统将生成的内容展示给用户。用户确认无误后, 点击“应用”将其保存到系统数据库中; 若不满意, 可要求 AI 重新生成。流程结束。

3.2.4. 项目 Dashboard 业务流程图

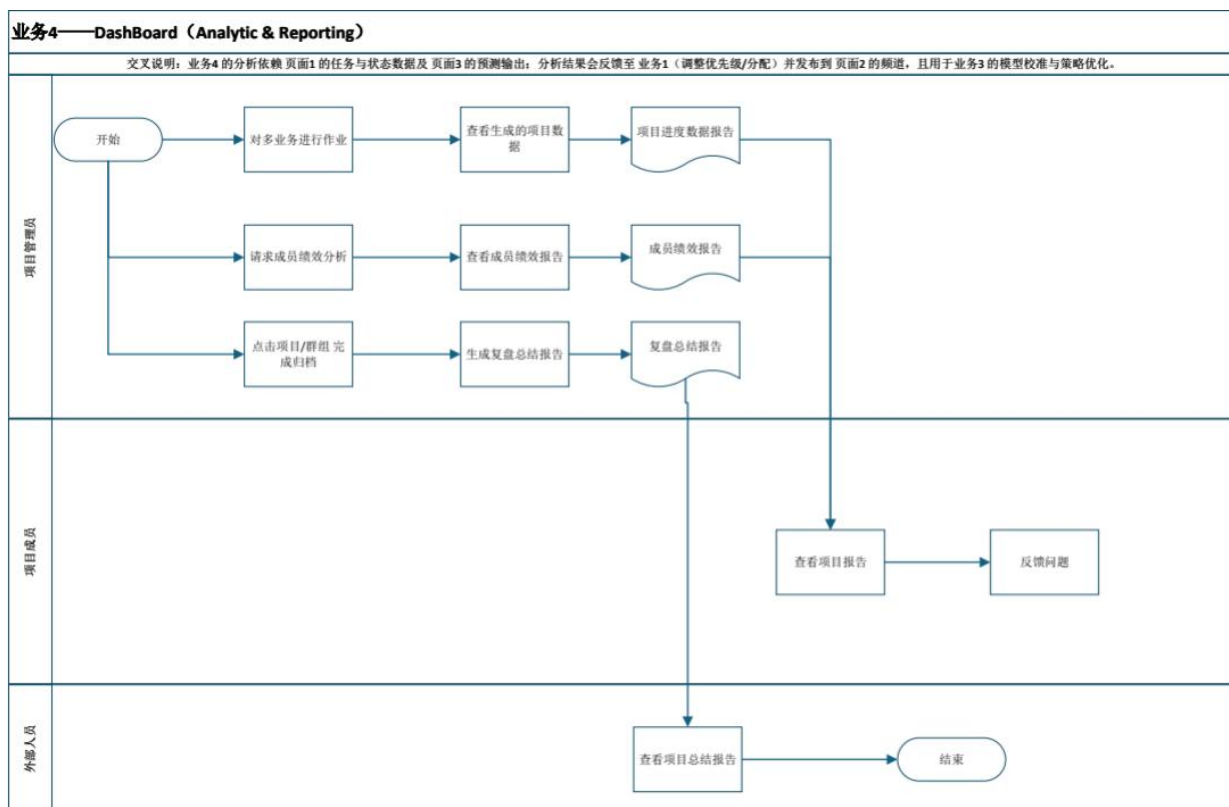


图 4 业务流程图——Dashboard

项目仪表盘业务流程：

1. 数据采集：系统定时从数据库中抽取项目进度、任务状态、工时记录等原始数据。
2. 数据处理与计算：后台统计服务对原始数据进行聚合计算，生成关键指标（如燃尽图数据、任务完成率、Bug 修复率）。
3. 可视化渲染：前端组件根据计算结果，动态渲染各种图表（如饼图、柱状图、趋势图）。
4. 用户查看与交互：项目经理或管理层访问仪表盘页面，查看实时数据，并可进行筛选（按时间、按人员）。系统根据用户的筛选条件实时刷新图表展示。流程结束。

4. 数据流程分析(张运鹏、张怡博)

4.1. 分层数据流图 (DFD)

4.1.1. 顶层数据流图

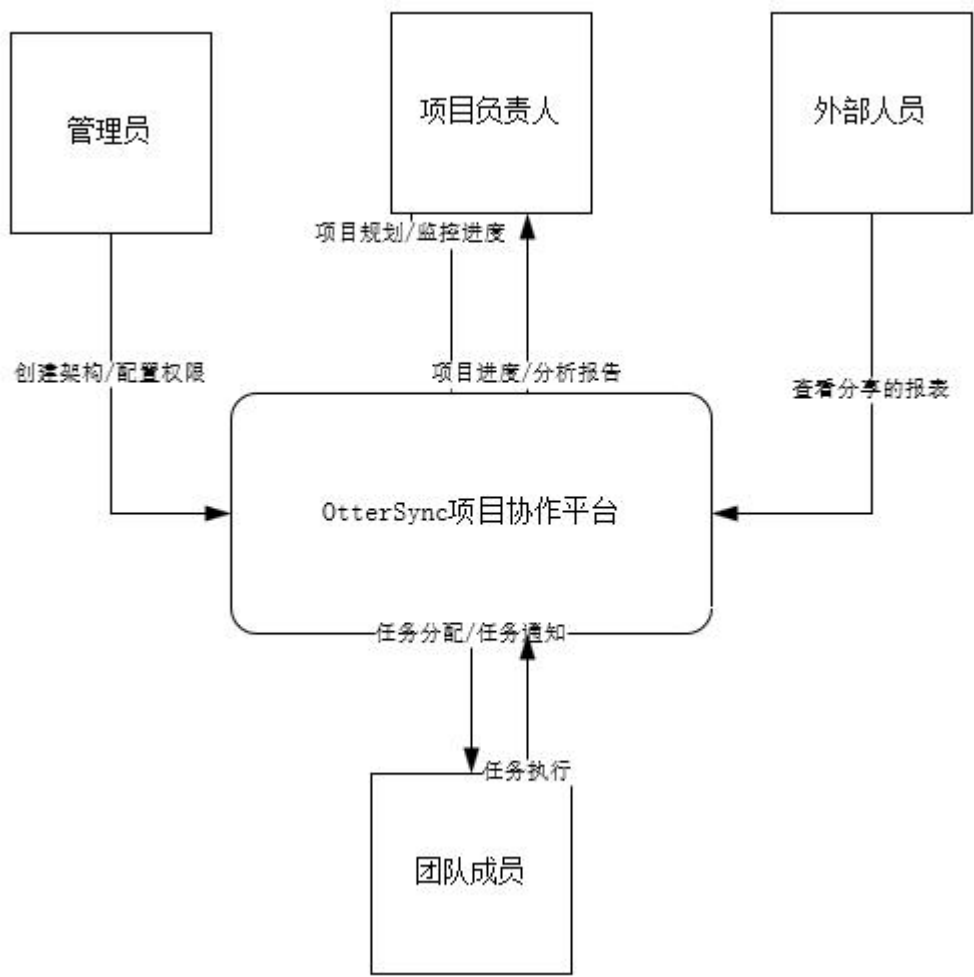


图 5 顶层数据流图

顶层数据流图描述:

系统为用户（开发、测试、PM）提供任务管理、协作和报表服务。外部实体包括“普通用户”、“项目经理”和“系统管理员”。

1. 用户向系统输入登录信息、任务详情、状态更新指令。
2. 系统管理员向系统输入配置信息和用户数据。
3. 系统处理后，向用户反馈任务列表、通知提醒和可视化报表。系统与各外部实体之间通过数据流实现了信息的交互与闭环。

4.1.2. 第一层数据流图

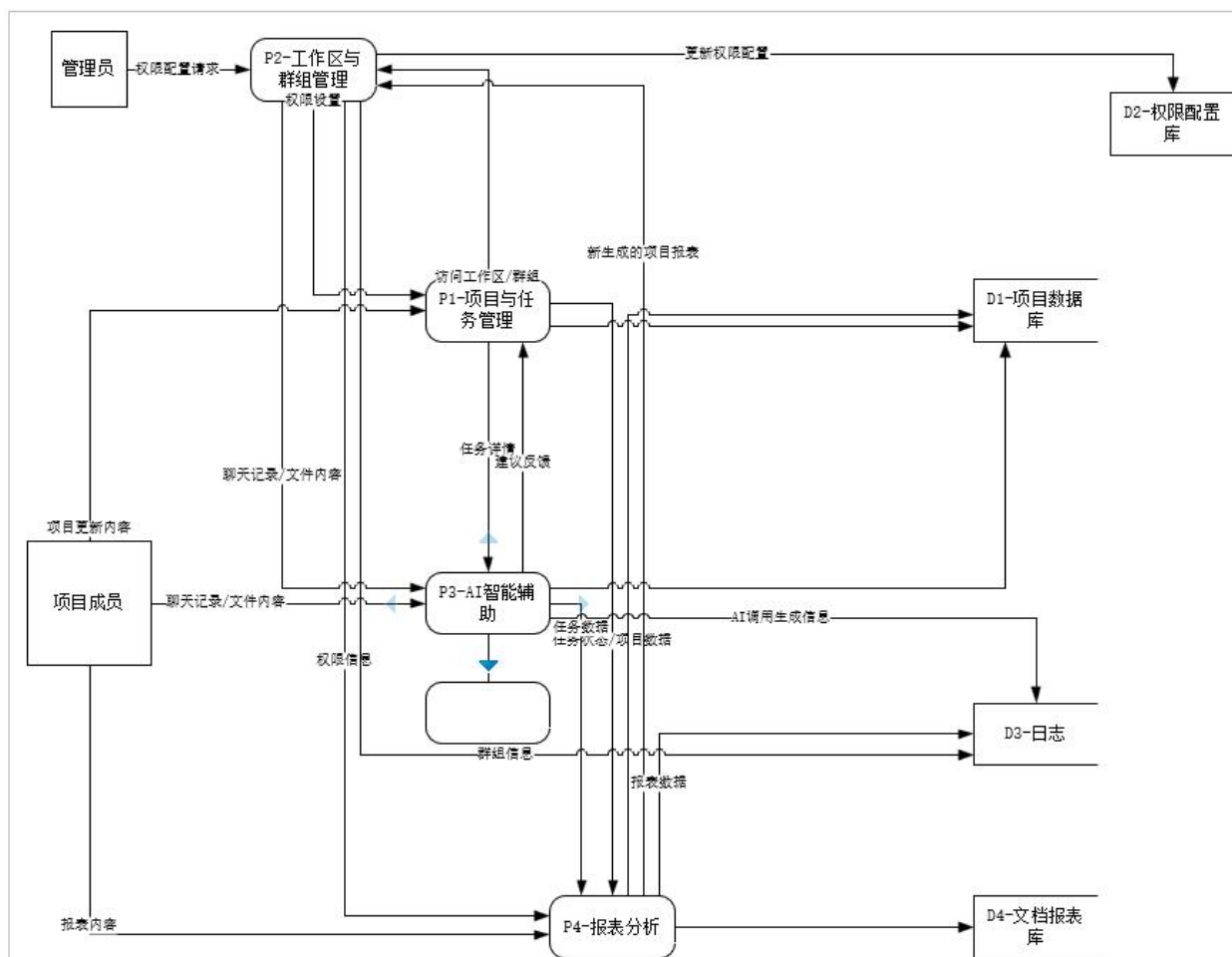


图 6 第一层——主流程

第一层数据流图描述:

系统主流程被分解为四个主要处理功能:

P1 项目管理：处理项目的创建、配置及 Sprint 规划。

P2 工作区与群组：处理用户认证、组织架构及权限校验。

P3 AI 智能辅助：处理自然语言请求，提供智能化建议。

P4 仪表盘统计：处理数据统计与报表生成。

各功能模块之间通过数据存储（如 D1 项目库、D2 任务库）进行数据交换。

4.1.3. 第二层数据流图

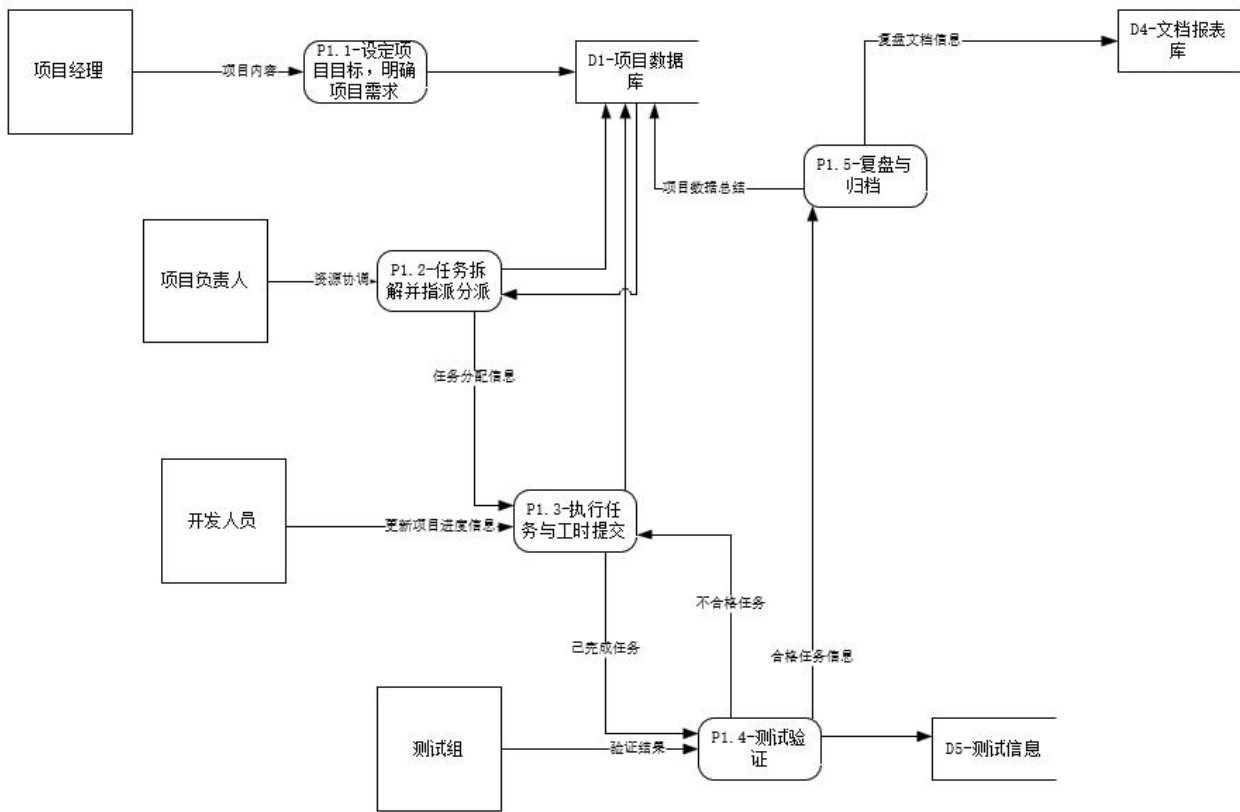


图 7 第二层——P1-项目管理

P1 项目管理数据流描述:

- P1.1 维护项目信息:** 接收 PM 的项目配置请求, 更新 D1 项目信息表。
- P1.2 规划 Sprint:** PM 从 Backlog 拖拽任务, 系统将任务关联关系写入 D3 迭代信息表。
- P1.3 任务流转:** 接收用户的状态变更指令, 校验 workflow 规则后, 更新 D2 任务信息表。

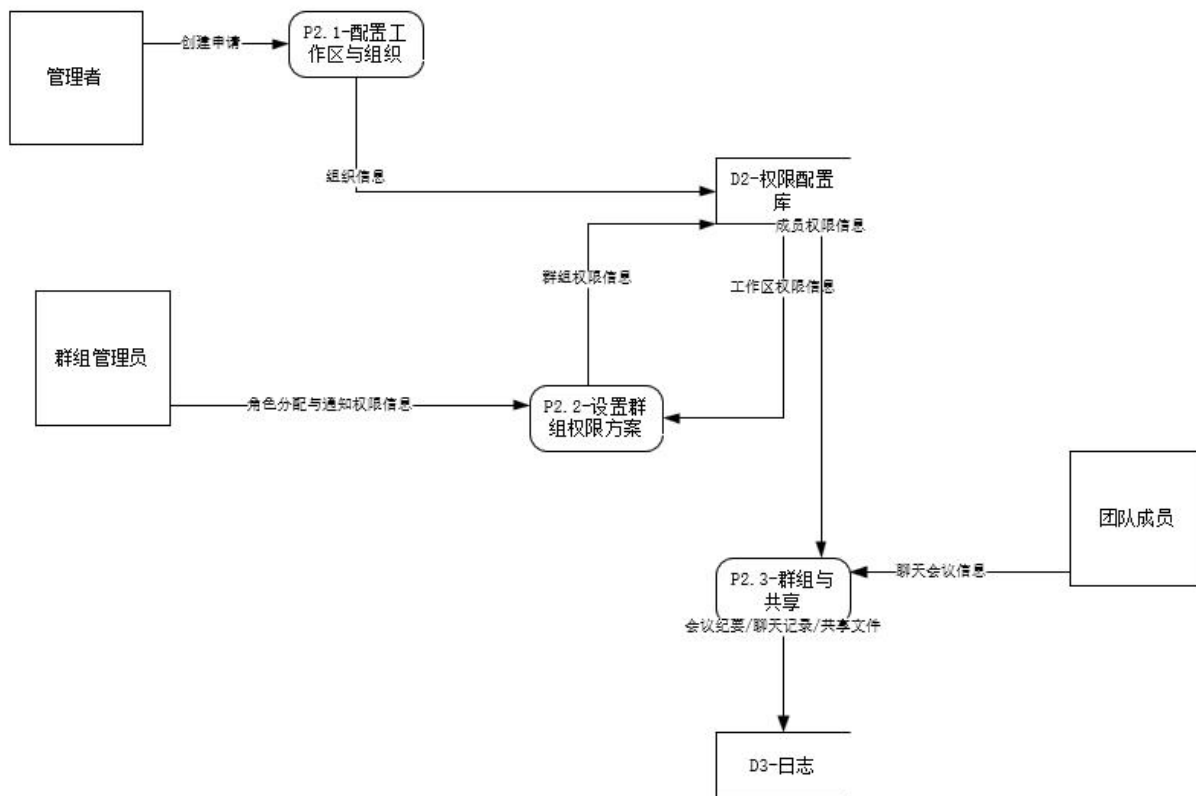


图 8 第二层——P2-工作区与群组

P2 工作区与群组数据流描述：

P2.1 用户认证：接收登录凭据，查询 D4 用户信息表，返回 Token。

P2.2 权限管理：接收管理员的授权指令，更新 D5 权限映射表。

P2.3 成员同步：处理新成员邀请，将新用户写入 D4 用户信息表，并发送通知。

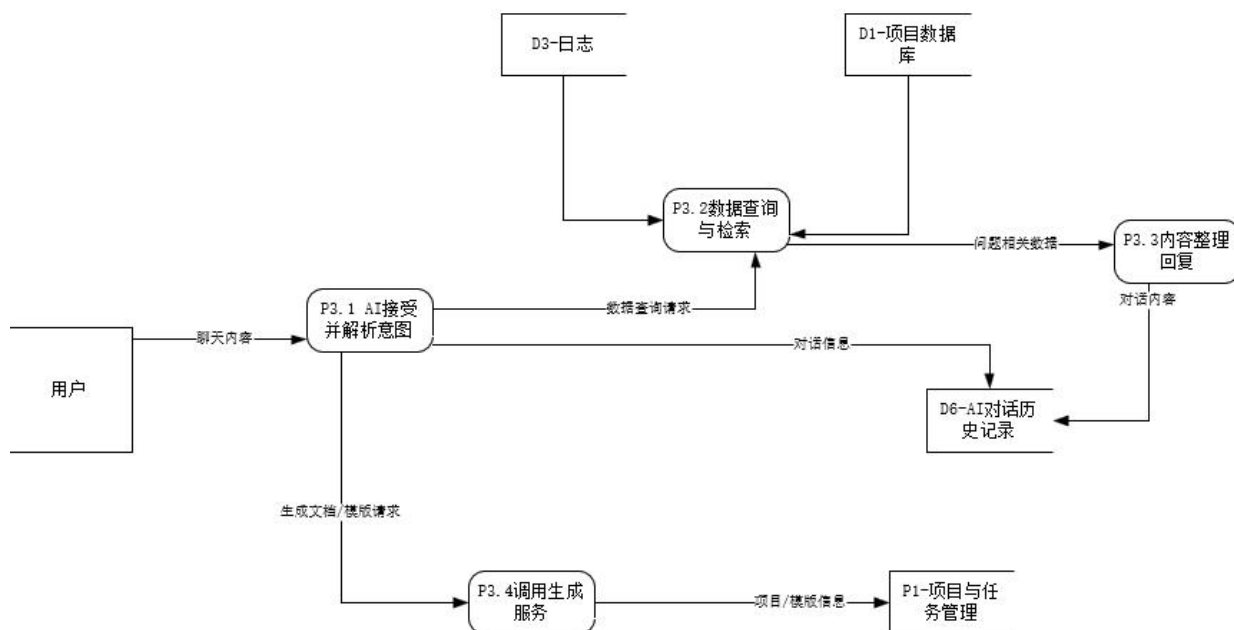


图 9 第二层——P3-AI 智能辅助

P3 AI 智能辅助数据流描述：

P3.1 语料预处理：读取任务描述（来自 D2），进行清洗和格式化。

P3.2 模型推理：将处理后的文本发送给 AI 模型，获取生成结果。

P3.3 结果应用：将 AI 生成的建议（如测试用例）回写到 D2 任务信息表或 D6 评论表中。

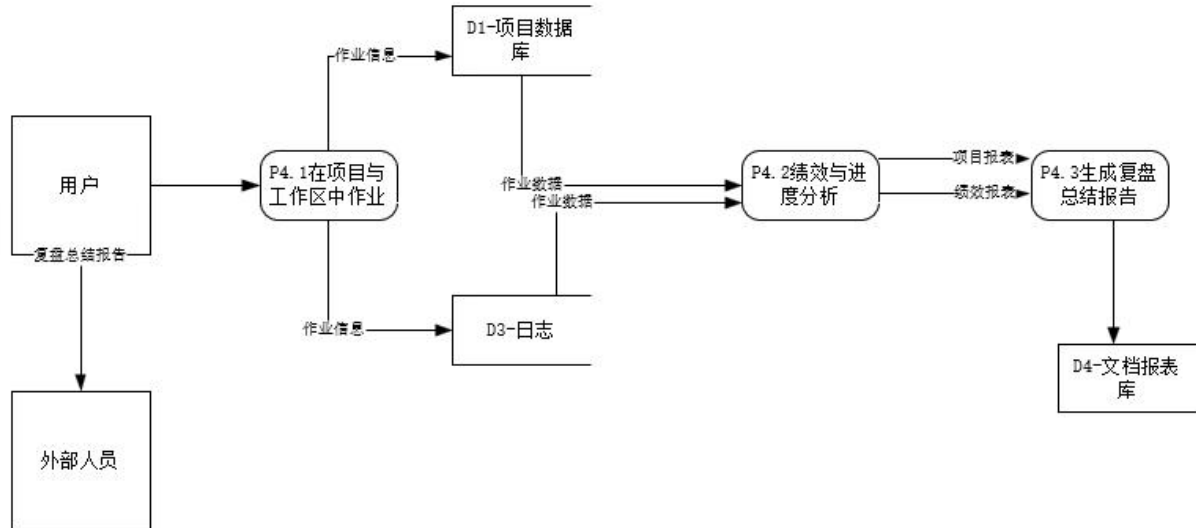


图 10 第二层——P4-仪表盘

P4 仪表盘数据流描述：

P4.1 指标计算：定期扫描 D2 任务表和 D3 迭代表，计算燃尽率和完成度。

P4.2 报表生成：根据计算结果生成 JSON 格式的图表数据。

P4.3 视图展示：响应前端请求，将报表数据发送给用户的浏览器进行渲染。

5. .数据汇总分析(张运鹏)

5.1. E-R 图设计 (概念模型)

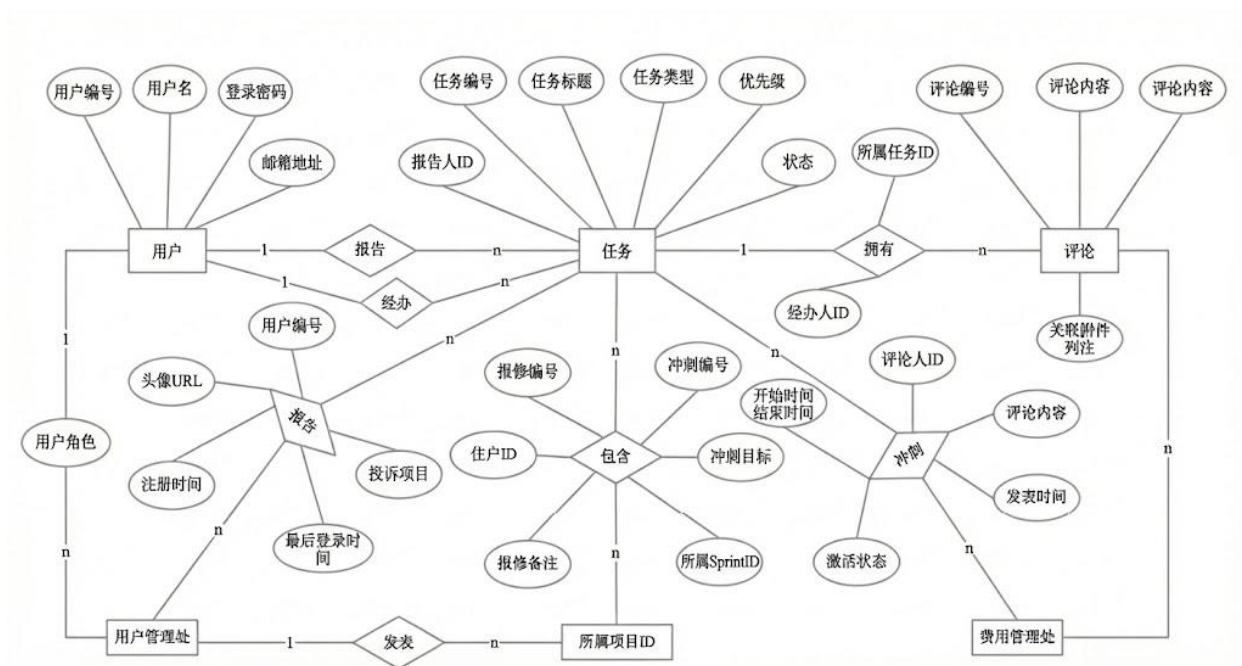


图 11 E-R 图

5.2. 关系表

本系统的关系模式定义如下（下划线标示主键，斜体标示外键）：

用户(User) (用户编号, 用户名, 登录密码, 邮箱地址, 头像 URL, 注册时间, 最后登录时间, 状态, 用户角色)

项目(Project) (项目编号, 项目名称, 项目代号, 项目描述, *创建人 ID*, 创建时间, 状态)

任务(Task) (任务编号, 任务标题, 任务描述, 任务类型, 优先级, 任务状态, 故事点, 创建时间, 更新时间, *经办人 ID*, *报告人 ID*, *所属项目 ID*)

评论(Comment) (评论编号, 评论内容, 发表时间, *所属任务 ID*, *评论人 ID*)

工时记录(WorkHourLog) (记录编号, 工时, 日期, *任务编号*, *开发人员 ID*)

项目成员关联(ProjectMember) (项目编号, 用户编号, 项目内角色, 加入时间)

权限映射(Permission) (权限编号, 权限名称, 权限描述, 适用角色 ID)

5.3. 数据字典

本数据字典对 OtterSync 系统中的主要数据项、数据结构及数据流进行了详细定义，旨在规范系统设计中的数据标准，确保各模块间的数据一致性。

5.3.1. 数据项定义

表 1 数据项条目——任务编号

| 数据元素条目 | |
|--|-------------------|
| 名称:任务编号 (Issue ID) 别名:issue_id 说明:系统中唯一标识一个任务、缺陷或用户故事的编号 类型:数字 (Long/BigInt) 长度:20 有关数据结构:任务 (Issue), 评论 (Comment), 附件 (Attachment) | 总编号:1-01 编号:01 |

本节定义了系统中最基础的原子数据元素。

表 2 数据项条目——任务优先级

| 数据元素条目 | |
|--|-------------------|
| 名称:优先级 (Priority) 别名:issue_priority 说明:标识任务处理的紧急程度，用于排序和资源分配 类型:字符 (String) 长度:10 有关数据结构:任务 (Issue) | 总编号:1-02 编号:02 |

表 3 数据项条目——任务状态

| 数据元素条目 | |
|-------------------------------------|-------------------|
| 名称:任务状态 (Status) 别名:issue_status | 总编号:1-03 编号:03 |

| | |
|---|--|
| 说明:标识任务在生命周期工作流中所处的 当前阶段 类型:字符 (String) 长度:20 有关数据结构:任务 (Issue) | |
|---|--|

表 4 数据项条目——用户角色

| 数据元素条目 | |
|---|-------------------|
| 名称:用户角色 (User Role) 别名:role_id 说明:定义用户在特定项目中的操作权限等级 类型:字符 (String) 长度:20 有关数据结构:用户 (User), 项目成员关联表 | 总编号:1-04 编号:04 |

表 5 数据项条目——项目代号

| 数据元素条目 | |
|--|-------------------|
| 名称:项目代号 (Project Key) 别名:project_key 说明:项目的短代码, 大写字母组成, 作为对该项目下所有任务编号的前缀 (如 "OTTER-101") 类型:字符 (String) 长度:10 有关数据结构:项目 (Project), 任务 (Issue) | 总编号:1-05 编号:05 |

表 6 数据项条目——故事点

| 数据元素条目 | |
|-----------------------|----------|
| 名称:故事点 (Story Points) | 总编号:1-06 |

| | |
|--|-------|
| 别名:estimate_points 说明:敏捷开发中用于评估任务复杂度或工 作量的相对单位 类型:数字 (Integer) 长度:3 有关数据结构:任务 (Issue) | 编号:06 |
|--|-------|

5.3.2. 数据结构定义

本节定义了系统中复合的数据对象及其组成属性。

表 7 数据结构条目——任务 (Issue)

| 数据结构条目 | |
|---|-------------------|
| 名称:任务 别名:Task 结构: <div> 任务编号 任务标题 任务描述 任务类型(Story/Task/Bug) 优先级 状态 经办人 ID 报告人 ID 故事点 创建时间 更新时间 所属项目 ID 所属 SprintID </div> | 总编号:2-01 编号:01 |

表 8 数据结构条目——用户 (User)

| 数据结构条目 | |
|--|-------------------|
| 名称:用户 别名:User 结构: 用户编号 用户名 登录密码(Hash 加密) 邮箱地址 头像 URL 注册时间 最后登录时间 状态(激活/禁用) | 总编号:2-02 编号:02 |

表 9 数据结构条目——评论 (Comment)

| 数据结构条目 | |
|---|-------------------|
| 名称:评论 别名:Comment 结构: 评论编号 所属任务 ID 评论人 ID 评论内容 发表时间 关联附件列表 | 总编号:2-04 编号:04 |

5.3.3. 数据流定义

本节定义了数据在系统各模块间流动的路径和载体。

表 10 数据流条目——提交任务

| 数据流条目 | |
|--|--|
| <p>名称:提交任务</p> <p>说明:用户在前端通过“创建任务”模态框填写的表单数据流向后端服务</p> <p>数据流来源:用户 (User) - 前端界面</p> <p>数据流去向:任务管理模块 (P2) - 后端控制器</p> <p>包含的数据结构:</p> <ul style="list-style-type: none"> 任务标题 任务描述 优先级 经办人 任务类型 附件 | <p>总编号:3-01</p> <p>编号:01</p> <p>流通量:约 50-100 次/天</p> |

表 11 数据流条目——状态更新指令

| 数据流条目 | |
|---|--|
| <p>名称:状态更新指令</p> <p>说明:用户在看板视图拖拽卡片或在详情页切换状态时触发的数据请求</p> <p>数据流来源:用户 (User) - 看板/详情页</p> <p>数据流去向:工作流引擎 (P2.2)</p> <p>包含的数据结构:</p> <ul style="list-style-type: none"> 任务编号 原状态 ID 目标状态 ID 操作人 ID 操作时间 | <p>总编号:3-02</p> <p>编号:02</p> <p>流通量:约 200+ 次/天</p> |

表 12 数据流条目——数据流条目

| 数据流条目 | |
|---|---|
| 名称:迭代报告数据 说明:系统根据任务历史记录计算出的当前冲刺统计信息，用于前端渲染图表 数据流来源:报表统计模块 (P4) 数据流去向:项目经理 (PM) - 报表界面 包含的数据结构: 用户 ID 总故事点数 已完成点数 剩余点数 燃尽图坐标序列 未完成任务列表 | 总编号:3-03 编号:03 流通量:低频(主要在 Sprint 结束时查询) |

5.3.4. 数据存储

本节定义了系统数据流程图中涉及的数据存储文件，它们是数据在系统内停留和保存的地方，对应数据库中的表或文件集合。

表 13 项目信息库

| 数据存储条目 | |
|--|--|
| 名称:项目信息库 说明:存放所有项目的基础配置信息、描述及状态 结构: 项目 ID 项目名称 项目代号 项目描述 创建人 ID | 总编号:4-01 编号:01 有关的数据流: P1.1->D1 D1->P1.2 D1->P4.1 信息量:50 份/年 有无立即查询:有 |

| | |
|------|--|
| 创建时间 | |
| 状态 | |

表 14 任务信息库

| 数据存储条目 | |
|--|--|
| 名称:任务信息库 说明:系统的核心存储，记录所有的任务、缺陷、用户故事及其流转状态 结构: 任务 ID 标题 描述 优先级 状态 经办人 ID 所属项目 ID 所属迭代 ID 故事点 | 总编号:4-02 编号:02 有关的数据流: P1.3->D2 P3.3->D2 D2->P4.1 D2->P3.1 信息量:1000 份/月 有无立即查询:有 |

表 15 用户与权限库

| 数据存储条目 | |
|--|--|
| 名称:用户与权限库 说明:存储用户的身份认证信息、个人资料及权限配置 结构: 用户 ID 用户名 密码 Hash 邮箱 角色 ID 所在用户组 ID | 总编号:4-03 编号:03 有关的数据流: P2.3->D4 D4->P2.1 D4->P2.2 信息量:50 份/月 有无立即查询:有 |

5.3.5 处理过程

表 16 任务流转处理

| 处理过程条目 | |
|---|-------------------|
| 名称:任务流转处理 输入:开发人员/项目经理 输出:任务信息库 (D2) 处理:用户发起任务状态变更请求，系统校验当前状态流转是否符合 workflow 规则，若符合则更新任务状态及时间。 | 总编号:5-01 编号:01 |

本节对数据流程图中底层的处理逻辑进行说明。

表 17 用户身份认证

| 处理过程条目 | |
|---|-------------------|
| 名称:用户身份认证 输入:用户 输出:用户 处理:用户提交账号密码，系统在数据库比对 Hash 值，验证通过后生成 Token 并返回给用户，同时记录登录日志。 | 总编号:5-02 编号:02 |

表 18 AI 内容生成

| 处理过程条目 | |
|--|-------------------|
| 名称:AI 内容生成 输入:用户 Prompt 输出:任务信息库 (D2) 处理:系统接收用户指令和上下文，调用外部大模型 API, 获取生成内容(如测试用例、代码建议)并存入任务记录。 | 总编号:5-03 编号:03 |

表 19 燃尽图计算

| 处理过程条目 | |
|---|-------------------|
| 名称:燃尽图计算 输入:任务信息库 (D2), 迭代冲刺库 (D3) 输出:项目经理 (PM) 处理:系统根据当前迭代周期, 聚合每日的 剩余任务点数, 计算理想曲线与实际曲线数 据, 生成图表数据返回。 | 总编号:5-04 编号:04 |

5.3.6 外部实体

本节定义了与系统交互的外部对象。

表 20 开发人员

| 处理过程条目 | |
|---|------------------------------|
| 名称:开发人员 输入数据流: P1.3->开发人员 (任务更新反 馈) P3.3->开发人员 (AI 建议) P4.3->开发人员 (报表视图) 输出数据流: 开发人员->P1.3 (状态更新指 令) 开发人员->P2.1 (登录信息) 开发人员->P3.1 (AI 对话指 令) | 总编号:6-01 编号:01 个数:50 个 |

表 21 项目经理

| 处理过程条目 | |
|--|-----------------------------|
| 名称:项目经理 输入数据流: P4.1->项目经理 (项目统计报 表) | 总编号:6-02 编号:02 个数:5 个 |

| | |
|--|--|
| <p>P4.2->项目经理 (进度风险预警)输出数据流:</p> <p>项目经理->P1.1 (项目创建配置)</p> <p>项目经理->P1.2 (Sprint 规划指令)</p> | |
|--|--|

表 22 系统管理员

| 处理过程条目 | |
|--|--|
| <p>名称:系统管理员</p> <p>输入数据流:</p> <p>P2.2->系统管理员 (权限操作反馈)</p> <p>输出数据流:</p> <p>系统管理员->P2.2 (权限变更指令)</p> <p>系统管理员->P2.3 (组织架构调整)</p> | <p>总编号:6-03</p> <p>编号:03</p> <p>个数:2 个</p> |

6. 系统功能及其划分（张凯恒）

6.1. 系统功能结构图

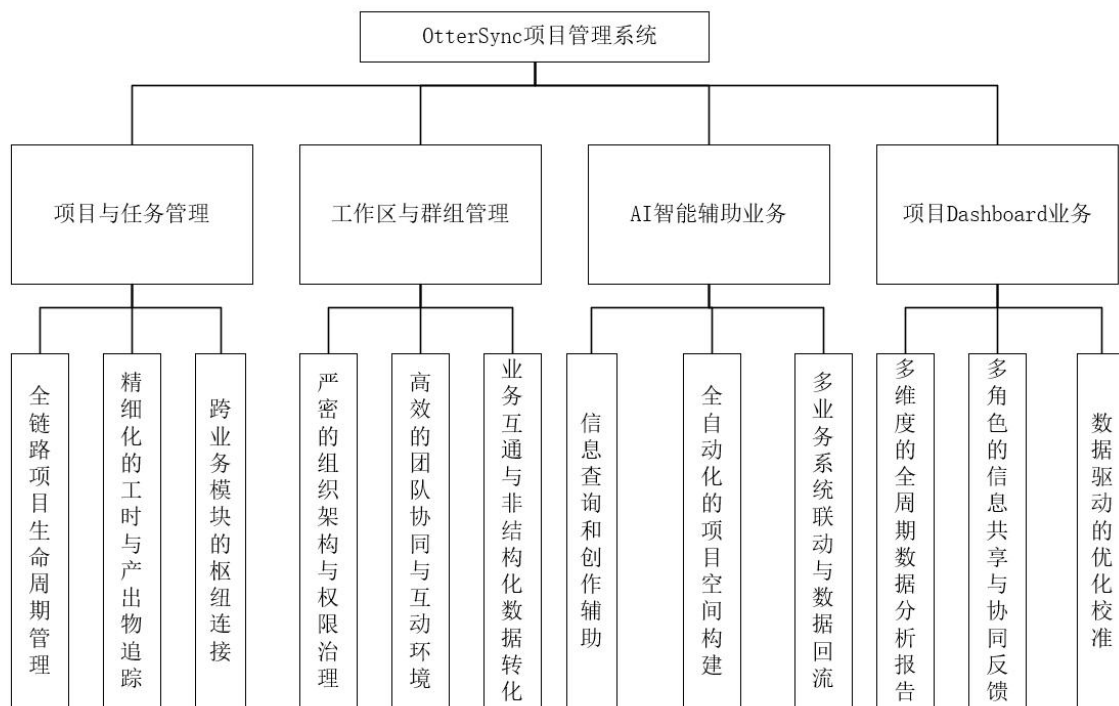


图 12 系统功能结构图

系统工程结构图：

OtterSync 项目管理系统主要由项目与任务管理系统、工作区与群组管理系统、AI 智能辅助业务系统、项目 Dashboard 业务系统组成。其中项目与任务管理系统包括全链路项目生命周期管理、精细化的工时与产出物追踪、跨业务模块的枢纽连接等功能，工作区与群组管理系统包括严密的组织架构与权限治理、高效的团队协作与互动环境、业务互通与非结构化数据转化等功能，AI 智能辅助业务系统包括信息查询和创作辅助、全自动化的项目空间构建、多业务系统联动与数据回流等功能，项目 Dashboard 业务系统包括多维度的全周期数据分析报告、多角色的信息共享与协同反馈、数据驱动的优化校准等功能。

7. 用例图与用例描述（王行健、张怡博）

7.1. 用例图

7.1.1 开发人员

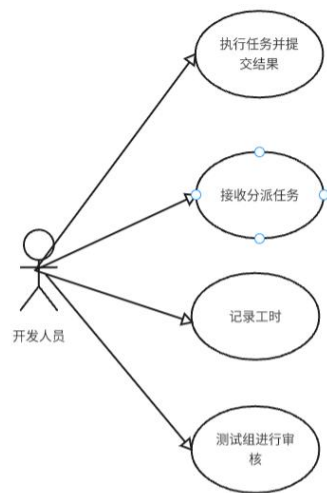


图 13 开发人员用例图

7.1.2 用户

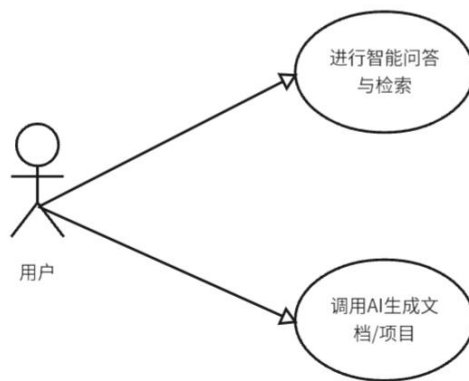


图 14 用户用例图

7.1.3 项目负责人

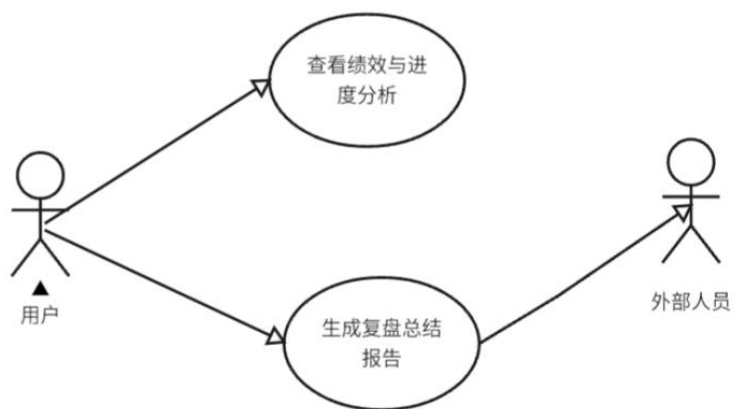


图 15 项目负责人用例图

7.2 用例分析

7.2.1 开发者完成开发任务

表 23 开发者完成任务用例描述

| | | | |
|--|--|---------------------|--|
| 用例名称 | 开发人员完成任务 | | |
| 参与者 | 开发人员（主要参与者），测试组（次要参与者） | | |
| 前置条件 | 任务已由项目经理拆解并分派到开发人员，开发人员已登录并获得相应权限，相关需求/工单和代码仓库可访问 | | |
| 后置条件 | 任务状态更新为“进行中/已提交”；工时记录被系统保存并计入项目工时统计；若提交了代码则触发构建/集成流程 | | |
| 主事件流 | | | |
| 1. 开发人员接收分派的任务并查看任务详情（目标、期限、验收标准）。 2. 开发人员在本地实现功能、编写或更新单元测试并在本地验证通过。 3. 开发人员将变更提交到代码仓库并发起合并请求或提交到指定分支。 4. 开发人员在系统中填写并提交当次工作内容与耗时（工时）。 | | 5. 系统记录工时并将工时计入对应任务 | |
| 备选事件流 | | | |
| 1a. 任务描述不清或缺少依赖：开发人员向项目经理或任务提交者发起澄清请求，暂停后续实现。项目经理补充说明或调整任务，流程返回第 2 步。 5a. 工时记录超出审批阈值或异常：系统标记为需审批，项目经理审核并确认或要求补充说明。 1b. 任务被取消或重规划：项目经理取消/重分配任务，系统更新状态，开发人员停止当前工作。 | | | |

7.2.2 智能问答与检索

表 24 用户智能问答与检索用例描述

| | |
|------|-------------------------------|
| 用例名称 | 用户进行智能问答与检索 |
| 参与者 | 用户（主要参与者）；系统管理员/知识库维护者（次要参与者） |

| | |
|---|--|
| 前置条件 | 用户已登录或具备访问权限；系统接入所需知识库/索引；问答服务或检索服务已启动并可用 |
| 后置条件 | 返回给用户的答案或检索结果被呈现并记录；检索/问答日志与用户反馈被保存；必要时触发人工工单或知识库更新流程 |
| 主事件流 | |
| 1. 用户在界面输入问题或查询并提交。 | 2. 系统校验用户身份和访问权限（若有访问控制） 3. 系统对输入做预处理（分词、意图识别、实体抽取）； 4. 系统在知识库/索引中检索相关文档/片段，获取候选证据； 5. 系统按排序规则筛选候选结果；若配置生成能力，则基于候选证据调用生成模型融合并输出答案； 6. 系统将答案（含引用/来源/置信度）呈现给用户，并提供相关链接或扩展阅读； |
| 备选事件流 | |
| 2a. 非法或未授权用户：系统提示身份或权限错误，记录并结束用例。 3a. 输入不完整或歧义：系统提示补充信息或给出澄清问题，用户补充后返回主流第3步。 5a. 无匹配或检索结果：系统提示无相关内容，并提供相近建议查询、人工提交选项或创建问题工单。 5b. 检索/生成超时或服务不可用：系统返回错误提示并按降级策略（如仅返回原始索引片段或排队重试）。 6a. 部分内容因权限或敏感性受限：系统屏蔽敏感部分并提示用户查看申请方式或提交审批。 | |

7.2.3 绩效查看与进度分析

表 25 绩效查看与进度分析用例描述

| | |
|------|---|
| 用例名称 | 查看绩效与进度分析 |
| 参与者 | 用户（主要参与者）；系统管理员/数据维护者（次要参与者）；外部人员（次要参与者） |
| 前置条件 | 用户已登录并具有查看权限；项目/任务与工时、测试、里程碑等数据已被系统采集并可访问；分析/报表服务已部署 |
| 后置条件 | 系统向用户呈现基于实时或近实时数据的绩效与进度分析视图；用户的查询与筛选条件被记录；如触发告警或发现偏差则生成通知或创建待办项 |
| 主事件流 | |

| | |
|--|--|
| <p>1. 用户打开“绩效与进度分析”页面并选择范围（项目/时间区间/团队）。</p> <p>5. 用户查看、筛选或导出分析结果；可保存为个人视图或共享链接</p> | <p>2. 系统校验用户权限并加载所选范围的数据源。</p> <p>3. 系统聚合并计算关键指标（完成率、里程碑达成、工时消耗、缺陷趋势、进度偏差等）。。</p> <p>4. 系统生成可视化图表与表格（趋势图、甘特视图、告警列表等），并按用户配置展示过滤与钻取选项</p> <p>6. 系统记录查询日志与导出操作并在检测到风险时触发告警或建议措施。</p> |
| 备选事件流 | |
| <p>a. 权限不足：系统提示无权限并阻止访问，记录访问尝试。</p> <p>b. 数据不完整或延迟：系统提示数据缺失或更新时间，并提供最近可用时间点或建议刷新操作。</p> <p>c. 分析计算超时或失败：系统返回部分结果或降级视图，提示稍后重试或异步通知完成。</p> <p>d. 发现重大偏差/风险：系统自动生成告警并建议创建纠正任务，或将结果推送给项目负责人。</p> | |

7.2.4 总结报告生成

表 26 总结报告生成用例描述

| | |
|------|---|
| 用例名称 | 生成复盘总结报告 |
| 参与者 | 用户（主要参与者）；外部人员/审阅者（次要参与者）；项目管理员（次要参与者） |
| 前置条件 | 项目或迭代已达交付/结束节点；相关交付物、日志、会议纪要、缺陷与度量数据已归档；用户具有生成或发起复盘权限 |
| 后置条件 | 生成复盘草稿并保存为文档记录；草稿被发送给外部人员或审阅者进行评审；最终复盘报告归档至知识库并生成后续改进任务 |
| 主事件流 | |

| | |
|---|---|
| <p>1. 用户在“生成复盘报告”入口选择项目/阶段与时间范围，并选择报告模板或定制要点。</p> <p>6. 用户查看、筛选或导出分析结果；可分享给外部人员</p> | <p>2. 系统校验权限并收集关联数据（进度、工时、缺陷、变更日志、测试结果、会议纪要等）。</p> <p>3. 系统基于模板聚合数据并生成报告草稿（可采用规则填充或调用 AI 自动摘要/撰写）。</p> <p>4. 系统对生成内容进行质量校验与敏感信息检测，并附上数据来源与关键证据。</p> <p>5. 系统将复盘报告归档</p> |
| 备选事件流 | |
| <p>2a. 关联数据不全或部分资料缺失：系统提示缺项并允许用户补充或选择降级模板，生成可能以可用数据为限。</p> <p>4a. 敏感或受限信息存在：系统自动脱敏或标注需人工审查，可能中止自动生成并进入人工处理</p> <p>4b. 自动生成结果质量不佳：用户选择手动编辑或请求多版本/重生成；系统保存各版本并记录变更历史。</p> <p>5a. 报告归档失败或存储受限：系统提示错误并重试或要求另存至指定位置。</p> <p>6a 外部人员不可见：系统反馈外部人员信息给用户，用户调整文件访问权限。</p> | |

8. 活动图、状态图及相关图的描述（张怡博）

8.1. 任务处理活动图

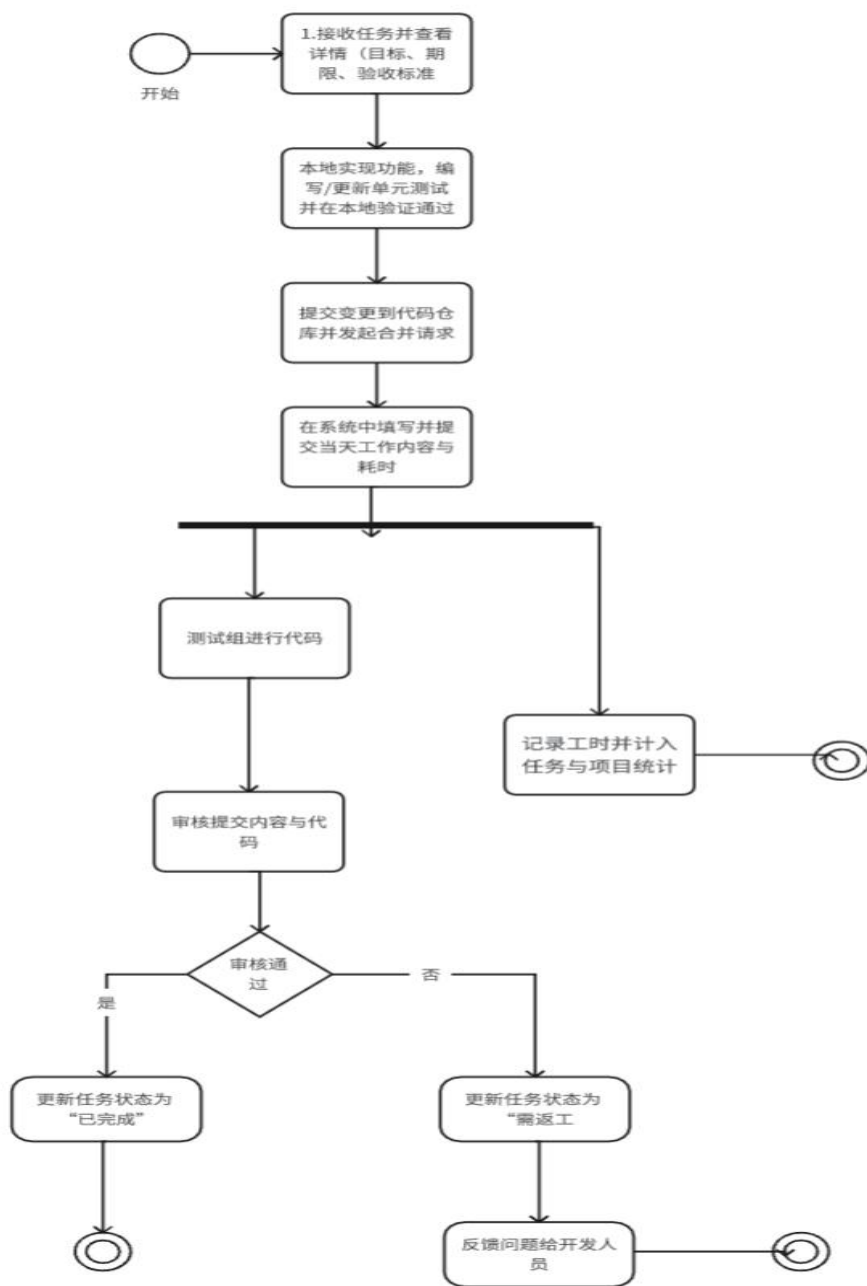


图 16 开发人员项目管理活动图

开发人员项目管理活动图分析：

- a) 任务分配系统
- b) 开发人员对任务进行代码开发，并将功能代码发送给测验组进行验证，若正确则提交代码，若错误则返回开发人员进行修改
- c) 开发人员记录工时计入项目统计，并更新项目任务状态

8.2. 任务生命周期状态图 (State Diagram)

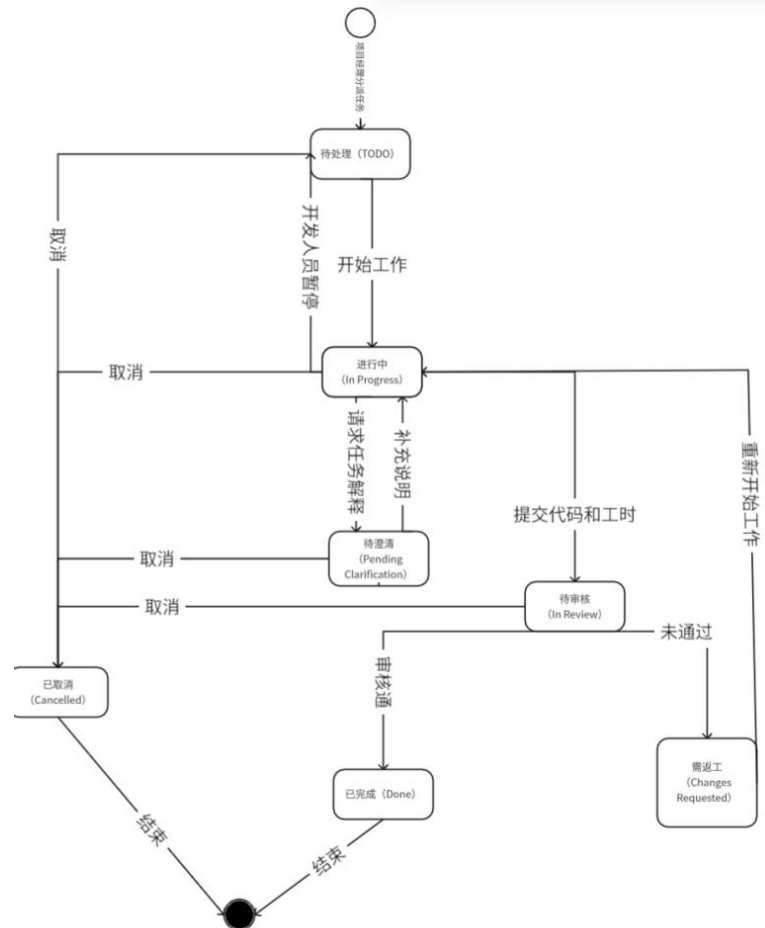


图 17 开发人员提交代码状态图：

开发人员提交代码状态图分析：

- a) 触发 “检测系统空闲”：
- b) 系统空闲 → 进入「进入系统」状态；
- c) 系统不空闲 → 触发 “结束等待” → 进程终止。
- d) 进入「登记住户信息中」状态 → 触发 “信息不完整” → 重新回到「登记住户信息中」状态
- e) 完成信息登记 → 进入「系统校验信息」状态：
- f) 触发 “信息已存在” → 进入「登记失败」状态；

g) 触发 “信息不存在” → （流程待补充后续操作）

9. 类图、类的关系及相关描述（张怡博）

9.1 项目管理系统类图分析

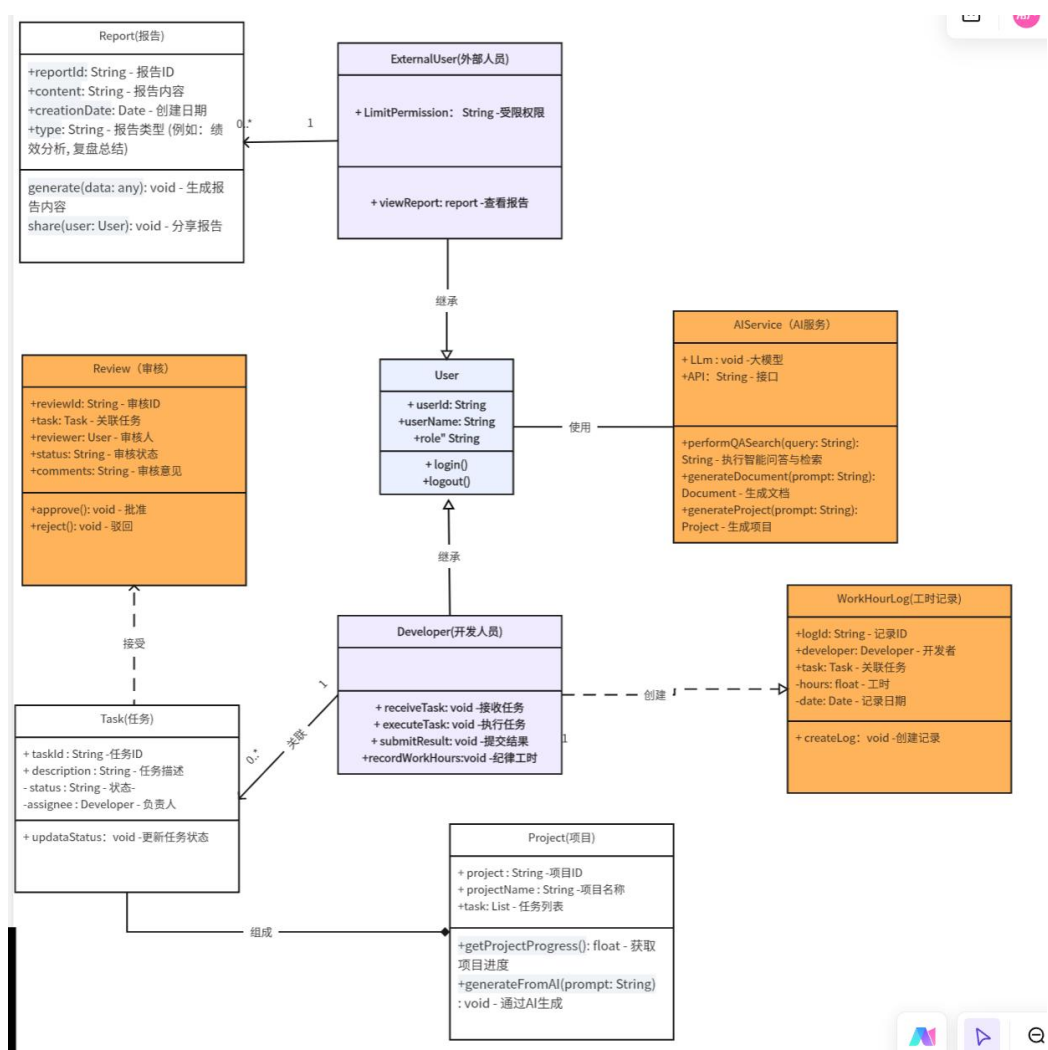


图 18 项目管理系统类图

该类图核心围绕用户与项目/任务/工时的管理，并集成了 AI 服务与复盘/报告、审核程。主要类包括：

User、ExternalUser、Developer（继承自 User）、Task、Project、WorkHourLog、AIService、

Report、Review。

User 为基类，Developer 与 ExternalUser 继承自 User（表示不同角色与权限），Developer 可接收并执行 Task（Task 与 Developer 通过 assignee 关联，通常为 Task 指向单个负责人，但可扩展为多人协作）。

Project 由若干 Task 组成（组合关系，Project 1 对 0..* Task），Project 可调用 AIService 生成概要或文档（依赖关系），User/Developer 会使用 AIService 提供的问与生成接口以支持检索与内容生成；

开发人员在执行任务时会创建 WorkHourLog（Developer 1 对 0..* WorkHourLog，WorkHourLog 关联到具体 Task，用于记录工时并可作为审计），而 Task 的提交会产生 Review（Review 与 Task 一一关联，且由某个 User/Reviewer 执行审核，审核结果驱动 Task 状态变更如通过或需返工）。

系统的 Report 实体表示可生成的报告（例如绩效或复盘），Report 与 ExternalUser 存在查看/共享关系（ExternalUser 具有限制权限，可查看或被授予访问特定 Report），同时 Report 可由系统或 AIService 生成并可被分享给 User；

总体上，实体之间以 Project→Task（组合）、Task↔Developer（指派/执行）、Developer→WorkHourLog（创建）、Task→Review（审核记录）、User/Developer→AIService（使用/依赖）、Report↔ExternalUser（查看/共享）等关系串联，形成任务分配、执行、工时记录、AI 辅助生成与审阅归档的完整协作流程。

9.2 类图图表描述分析

| User（用户） | 基础用户类 |
|----------|-------|
| 属性 | 说明 |

| | |
|---|---|
| <ul style="list-style-type: none"> ● <code>userId: String</code> ● <code>userName: String</code> ● <code>role: String</code> | 基础用户类，其他用户类型继承此类 (如 <code>Developer</code> 、 <code>ExternalUser</code>) 。 |
| 方法 | |
| <ul style="list-style-type: none"> ● <code>login()</code> ● <code>logout()</code> | |

| | |
|---|------------------------------|
| ExternalUser（外部人员） | 基础用户类- 继承自 <code>User</code> |
| 属性 | 说明 |
| <ul style="list-style-type: none"> ● <code>LimitPermission: String</code> （受限权限，如只读/查看特定报告 | 用于外部审阅或被授权查看报告的账户，权限受限。 |
| 方法 | |
| <ul style="list-style-type: none"> ● <code>viewReport(report: Report)</code> | |

| | |
|----------|-----|
| Task（任务） | 项目类 |
|----------|-----|

| 属性 | 说明 |
|---|--|
| <ul style="list-style-type: none"> ● taskId: String ● description: String ● status: String ● assignee: Developer （责任人 | 任务由 Project 组成的一部分；可被 0..* 个开发人员关联（通常为一个负责人，但可扩展为多人协作） |
| 方法 | |
| <ul style="list-style-type: none"> ● updateStatus(): void | |

| Project（项目 | 项目类 |
|---|-------------------------------|
| 属性 | 说明 |
| <ul style="list-style-type: none"> ● project: String（项目 ID） ● projectName: String ● task: List<Task>（任务列表） | 项目包含任务集合，并可调用 AIService 生成内容。 |
| 方法 | |
| <ul style="list-style-type: none"> ● getProjectProgress(): float — 获取项目进度 ● generateFromAI(prompt: String): void — 通过 AI 生成（如生成项目 | |

| WorkHourLog（工时记录） | 项目类 |
|---|--|
| 属性 | 说明 |
| <ul style="list-style-type: none"> ● logId: String ● developer: Developer ● task: Task ● hours: float ● date: Date | <p>由 Developer 创建，记录某任务的工时； 与 Task、Developer 关联。</p> |
| 方法 | |
| <ul style="list-style-type: none"> ● createLog(): void | |

| AIService（AI 服务） | 项目类 |
|--|---|
| 属性 | 说明 |
| <ul style="list-style-type: none"> ● Llm: void (表示大型模型，或模型实例) ● API: String（接口/配置） | <p>被 User/Developer/Project 等使用，提供 问答与生成能力。</p> |
| 方法 | |

- performQASearch(query: String): String — 执行智能问答与检索
- generateDocument(prompt: String): Document — 生成文档（类 Document 可映射为 Report 或独立实体）
- generateProject(prompt: String): Project — 生成项目草案/概要

10. 时序图、协作图及相关图的描述（张怡博）

10.1. 时序图

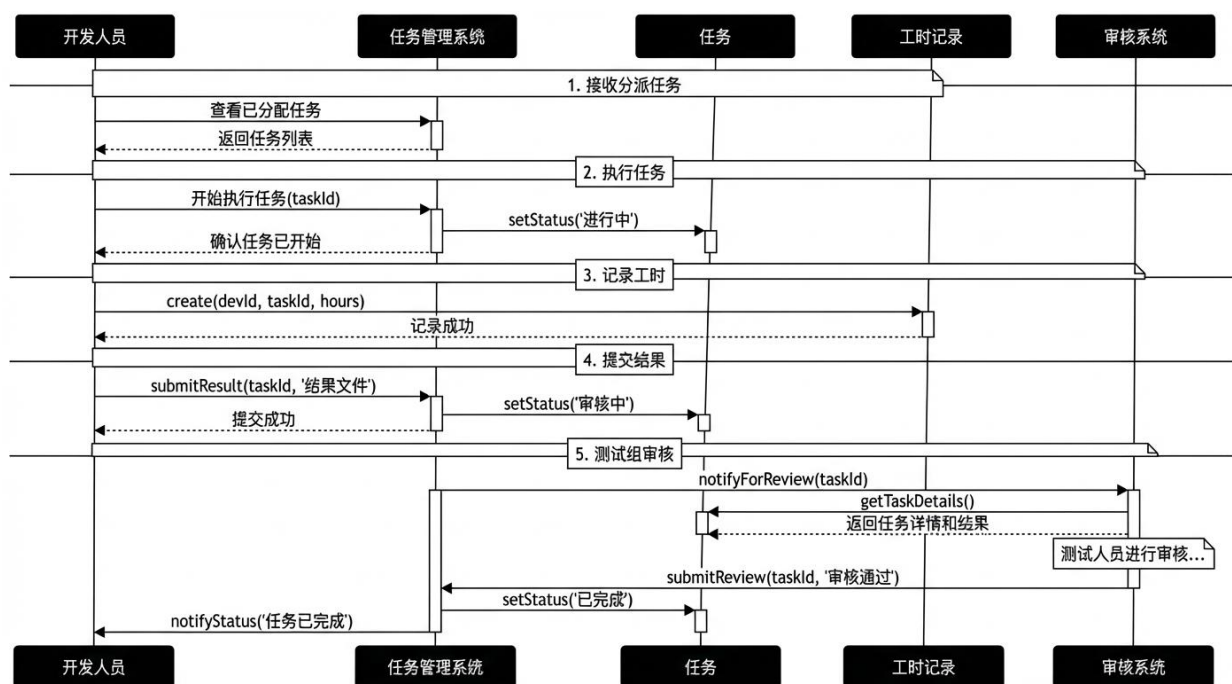


图 19 开发人员提交任务时序图

时序图描述;

1. 开发人员首先在任务管理系统中查看已分配的任务并获取任务列表后接收分派的任务；
2. 接着开发人员开始执行指定任务（传入 taskId），任务管理系统将该任务状态置为“进行中”并返回确认开发已开始执行；

3. 在执行过程中，开发人员向工时记录模块创建工时条目（`create(devId, taskId, hours)`），工时记录成功后返回记录成功信息；
4. 开发人员完成工作后向任务管理系统提交结果（`submitResult(taskId, "结果文件")`），系统将任务状态更新为“审核中”并返回提交成功；
5. 随后任务进入测试组/审核系统环节，任务或系统发出审核通知（`notifyForReview(taskId)`）
6. 审核系统调用 `getTaskDetails()` 获取任务详情与提交结果并返回相关信息，测试人员或审阅者进行审核并通过后通过 `submitReview(taskId, "审核通过")` 将审核结果回传给任务管理系统
7. 任务管理系统接收到审核通过后将任务状态设置为“已完成”，并向开发人员发送通知（`notifyStatus("任务已完成")`）结束整个流程。