**工业软件后端 - 登录功能测试用例文档**

**文档信息**

**项目名称：**工业软件后端系统  
**功能模块：**用户登录功能  
**接口路径：**/auth/jsonLogin  
**请求方法：**POST  
**文档版本：**v1.0  
**创建日期：**2025-08-07  
**测试工具：**Apifox

**一、测试概述**

**1.1 测试目标**

验证用户登录功能的正确性、稳定性和安全性，确保系统能够正确处理各种登录场景。

**1.2 测试环境**

**Base URL：**http://localhost:8081  
**数据库：**MySQL 8.0  
**缓存：**Redis  
**测试工具：**Apifox

**二、测试用例详细设计**

**2.1 功能测试用例**

**测试用例 TC001: 正常登录功能测试**

**用例ID：**TC001  
**用例名称：**正常登录功能测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**系统正常运行，数据库中有测试用户数据

**测试数据**

{ "username": "admin", "password": "123456", "verificationCode": "1234", "key": "testKey" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "操作成功", "data": { "userId": 1, "username": "admin", "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...", "permission": 1, "phone": "13800138000" } }

**测试用例 TC002: 用户名错误测试**

**用例ID：**TC002  
**用例名称：**用户名错误测试  
**测试类型：**异常测试  
**优先级：**高  
**前置条件：**系统正常运行

**测试数据**

{ "username": "wronguser", "password": "123456", "verificationCode": "1234", "key": "testKey" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "用户名或密码错误", "data": null }

**测试用例 TC003: 密码错误测试**

**用例ID：**TC003  
**用例名称：**密码错误测试  
**测试类型：**异常测试  
**优先级：**高  
**前置条件：**系统正常运行

**测试数据**

{ "username": "admin", "password": "wrongpassword", "verificationCode": "1234", "key": "testKey" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "用户名或密码错误", "data": null }

**测试用例 TC004: 参数缺失测试**

**用例ID：**TC004  
**用例名称：**参数缺失测试  
**测试类型：**参数验证测试  
**优先级：**中  
**前置条件：**系统正常运行

**测试数据**

{ "username": "admin", "verificationCode": "1234", "key": "testKey" }

**预期结果**

**状态码：**400 或 200  
**响应格式：**JSON  
**响应内容：**返回参数验证错误信息

**用户管理功能测试用例文档**

**文档信息**

**项目名称：**工业软件后端系统  
**功能模块：**用户管理功能  
**接口路径：**/modUsers  
**请求方法：**GET/POST  
**文档版本：**v1.0  
**创建日期：**2025-08-07  
**测试工具：**Apifox

**一、测试概述**

**1.1 测试目标**

验证用户管理功能的正确性、稳定性和安全性，包括用户注册、用户信息查询、密码修改等功能。

**1.2 测试环境**

**Base URL：**http://localhost:8081  
**数据库：**MySQL 8.0  
**缓存：**Redis  
**测试工具：**Apifox

**二、测试用例详细设计**

**2.1 用户注册功能测试**

**测试用例 TC001: 正常用户注册测试**

**用例ID：**TC001  
**用例名称：**正常用户注册测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**系统正常运行，数据库连接正常

**测试数据**

{ "username": "testuser001", "password": "123456", "phone": "13800138001", "email": "test001@example.com" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "操作成功", "data": { "user": { "userId": 2, "username": "testuser001", "phone": "13800138001", "email": "test001@example.com", "permission": 0 } } }

**测试用例 TC002: 用户名重复注册测试**

**用例ID：**TC002  
**用例名称：**用户名重复注册测试  
**测试类型：**异常测试  
**优先级：**高  
**前置条件：**系统中已存在用户名为"admin"的用户

**测试数据**

{ "username": "admin", "password": "123456", "phone": "13800138002", "email": "test002@example.com" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "用户名已存在", "data": null }

**2.2 用户信息查询功能测试**

**测试用例 TC003: 获取当前用户信息测试**

**用例ID：**TC003  
**用例名称：**获取当前用户信息测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**用户已登录，有有效的token

**测试数据**

请求头：Authorization: Bearer {token}

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "操作成功", "data": { "userId": 1, "username": "admin", "phone": "13800138000", "email": "admin@example.com", "permission": 1 } }

**测试用例 TC004: 未登录获取用户信息测试**

**用例ID：**TC004  
**用例名称：**未登录获取用户信息测试  
**测试类型：**异常测试  
**优先级：**中  
**前置条件：**用户未登录或token无效

**测试数据**

请求头：无Authorization或无效token

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "获取用户信息失败", "data": null }

**2.3 密码修改功能测试**

**测试用例 TC005: 正常密码修改测试**

**用例ID：**TC005  
**用例名称：**正常密码修改测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**用户已登录，原密码正确

**测试数据**

{ "oldPassword": "123456", "newPassword": "newpassword123" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "密码修改成功", "data": null }

**测试用例 TC006: 原密码错误修改测试**

**用例ID：**TC006  
**用例名称：**原密码错误修改测试  
**测试类型：**异常测试  
**优先级：**高  
**前置条件：**用户已登录

**测试数据**

{ "oldPassword": "wrongpassword", "newPassword": "newpassword123" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "原密码错误", "data": null }

**三、测试数据准备**

**3.1 测试用户数据**

-- 测试用户数据（需要在数据库中准备） INSERT INTO mod\_users (username, password, permission, phone, email) VALUES ('admin', 'encoded\_password\_123456', 1, '13800138000', 'admin@example.com'), ('testuser', 'encoded\_password\_123456', 0, '13800138001', 'test@example.com');

**3.2 测试环境变量**

// Apifox环境变量配置 { "baseUrl": "http://localhost:8081", "loginToken": "", "testUsername": "admin", "testPassword": "123456", "newTestUsername": "testuser001" }

**项目管理功能测试用例文档**

**文档信息**

**项目名称：**工业软件后端系统  
**功能模块：**项目管理功能  
**接口路径：**/modProjects  
**请求方法：**GET/POST  
**文档版本：**v1.0  
**创建日期：**2025-08-07  
**测试工具：**Apifox

**一、测试概述**

**1.1 测试目标**

验证项目管理功能的正确性、稳定性和安全性，包括共享项目查询、私人项目查询、项目创建等功能。

**1.2 测试环境**

**Base URL：**http://localhost:8081  
**数据库：**MySQL 8.0  
**缓存：**Redis  
**测试工具：**Apifox

**二、测试用例详细设计**

**2.1 共享项目管理功能测试**

**测试用例 TC001: 获取共享项目分页列表测试**

**用例ID：**TC001  
**用例名称：**获取共享项目分页列表测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**系统正常运行，数据库中有共享项目数据

**测试数据**

{ "pageNum": 1, "pageSize": 10, "keyword": "" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "操作成功", "data": { "total": 5, "records": [ { "projectId": 1, "projectName": "测试项目1", "description": "这是一个测试项目", "createTime": "2025-08-07 10:00:00", "status": "active" } ] } }

**测试用例 TC002: 创建共享项目测试**

**用例ID：**TC002  
**用例名称：**创建共享项目测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**用户已登录，有创建项目权限

**测试数据**

{ "projectName": "新测试项目", "description": "这是一个新的测试项目", "projectType": "shared" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "项目创建成功", "data": { "projectId": 6, "projectName": "新测试项目", "description": "这是一个新的测试项目", "createTime": "2025-08-07 10:30:00" } }

**2.2 私人项目管理功能测试**

**测试用例 TC003: 获取私人项目分页列表测试**

**用例ID：**TC003  
**用例名称：**获取私人项目分页列表测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**用户已登录，数据库中有该用户的私人项目数据

**测试数据**

{ "pageNum": 1, "pageSize": 10, "keyword": "" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "操作成功", "data": { "total": 3, "records": [ { "projectId": 2, "projectName": "我的私人项目1", "description": "这是我的私人项目", "createTime": "2025-08-07 09:00:00", "status": "active" } ] } }

**测试用例 TC004: 创建私人项目测试**

**用例ID：**TC004  
**用例名称：**创建私人项目测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**用户已登录

**测试数据**

{ "projectName": "我的新私人项目", "description": "这是我的新私人项目", "projectType": "private" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "项目创建成功", "data": { "projectId": 7, "projectName": "我的新私人项目", "description": "这是我的新私人项目", "createTime": "2025-08-07 11:00:00" } }

**2.3 异常情况测试**

**测试用例 TC005: 未登录获取私人项目测试**

**用例ID：**TC005  
**用例名称：**未登录获取私人项目测试  
**测试类型：**异常测试  
**优先级：**中  
**前置条件：**用户未登录或token无效

**测试数据**

{ "pageNum": 1, "pageSize": 10, "keyword": "" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "用户未登录", "data": null }

**测试用例 TC006: 项目名称重复创建测试**

**用例ID：**TC006  
**用例名称：**项目名称重复创建测试  
**测试类型：**异常测试  
**优先级：**中  
**前置条件：**用户已登录，系统中已存在同名项目

**测试数据**

{ "projectName": "测试项目1", "description": "重复的项目名称", "projectType": "shared" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "项目名称已存在", "data": null }

**三、测试数据准备**

**3.1 测试项目数据**

-- 测试项目数据（需要在数据库中准备） INSERT INTO mod\_projects (project\_name, description, project\_type, create\_user\_id, status) VALUES ('测试项目1', '这是一个测试项目', 'shared', 1, 'active'), ('我的私人项目1', '这是我的私人项目', 'private', 1, 'active'), ('共享项目2', '另一个共享项目', 'shared', 2, 'active');

**3.2 测试环境变量**

// Apifox环境变量配置 { "baseUrl": "http://localhost:8081", "loginToken": "", "testUserId": 1, "testProjectName": "新测试项目" }

**权限管理功能测试用例文档**

**文档信息**

**项目名称：**工业软件后端系统  
**功能模块：**权限管理功能  
**接口路径：**/user  
**请求方法：**POST  
**文档版本：**v1.0  
**创建日期：**2025-08-07  
**测试工具：**Apifox

**一、测试概述**

**1.1 测试目标**

验证权限管理功能的正确性、稳定性和安全性，包括用户分页查询、用户权限修改、用户组织修改等功能。

**1.2 测试环境**

**Base URL：**http://localhost:8081  
**数据库：**MySQL 8.0  
**缓存：**Redis  
**测试工具：**Apifox

**二、测试用例详细设计**

**2.1 用户分页查询功能测试**

**测试用例 TC001: 正常获取用户分页列表测试**

**用例ID：**TC001  
**用例名称：**正常获取用户分页列表测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**系统正常运行，数据库中有用户数据

**测试数据**

{ "pageNum": 1, "pageSize": 10, "keyword": "" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "操作成功", "data": { "total": 5, "records": [ { "userId": 1, "username": "admin", "phone": "13800138000", "email": "admin@example.com", "permission": 1, "orgName": "技术部" } ] } }

**测试用例 TC002: 带关键字搜索用户测试**

**用例ID：**TC002  
**用例名称：**带关键字搜索用户测试  
**测试类型：**功能测试  
**优先级：**中  
**前置条件：**系统正常运行，数据库中有包含"admin"关键字的用户

**测试数据**

{ "pageNum": 1, "pageSize": 10, "keyword": "admin" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "操作成功", "data": { "total": 1, "records": [ { "userId": 1, "username": "admin", "phone": "13800138000", "email": "admin@example.com", "permission": 1, "orgName": "技术部" } ] } }

**2.2 用户权限修改功能测试**

**测试用例 TC003: 管理员正常修改用户权限测试**

**用例ID：**TC003  
**用例名称：**管理员正常修改用户权限测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**当前用户为管理员权限，目标用户存在

**测试数据**

{ "userId": "2", "permission": 1 }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "权限修改成功", "data": null }

**测试用例 TC004: 非管理员修改用户权限测试**

**用例ID：**TC004  
**用例名称：**非管理员修改用户权限测试  
**测试类型：**异常测试  
**优先级：**高  
**前置条件：**当前用户为非管理员权限

**测试数据**

{ "userId": "2", "permission": 1 }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "没有权限执行此操作", "data": null }

**测试用例 TC005: 用户ID格式错误测试**

**用例ID：**TC005  
**用例名称：**用户ID格式错误测试  
**测试类型：**异常测试  
**优先级：**中  
**前置条件：**当前用户为管理员权限

**测试数据**

{ "userId": "invalid\_id", "permission": 1 }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "用户ID格式错误", "data": null }

**2.3 用户组织修改功能测试**

**测试用例 TC006: 正常修改用户组织测试**

**用例ID：**TC006  
**用例名称：**正常修改用户组织测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**用户和组织都存在

**测试数据**

{ "userId": "2", "orgId": "3" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "用户组织修改成功", "data": null }

**测试用例 TC007: 用户不存在修改组织测试**

**用例ID：**TC007  
**用例名称：**用户不存在修改组织测试  
**测试类型：**异常测试  
**优先级：**中  
**前置条件：**目标用户不存在

**测试数据**

{ "userId": "999", "orgId": "3" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "用户不存在", "data": null }

**三、测试数据准备**

**3.1 测试用户数据**

-- 测试用户数据（需要在数据库中准备） INSERT INTO mod\_users (username, password, permission, phone, email) VALUES ('admin', 'encoded\_password\_123456', 1, '13800138000', 'admin@example.com'), ('user001', 'encoded\_password\_123456', 0, '13800138001', 'user001@example.com'), ('user002', 'encoded\_password\_123456', 0, '13800138002', 'user002@example.com');

**3.2 测试组织数据**

-- 测试组织数据（需要在数据库中准备） INSERT INTO organizations (org\_name, description) VALUES ('技术部', '负责技术开发'), ('产品部', '负责产品设计'), ('测试部', '负责质量测试');

**3.3 测试环境变量**

// Apifox环境变量配置 { "baseUrl": "http://localhost:8081", "adminToken": "", "userToken": "", "testUserId": "2", "testOrgId": "3" }

**文件管理功能测试用例文档**

**文档信息**

**项目名称：**工业软件后端系统  
**功能模块：**文件管理功能  
**接口路径：**/file  
**请求方法：**GET/POST  
**文档版本：**v1.0  
**创建日期：**2025-08-07  
**测试工具：**Apifox

**一、测试概述**

**1.1 测试目标**

验证文件管理功能的正确性、稳定性和安全性，包括子程序文件上传下载、仿真输入文件上传下载等功能。

**1.2 测试环境**

**Base URL：**http://localhost:8081  
**数据库：**MySQL 8.0  
**缓存：**Redis  
**测试工具：**Apifox

**二、测试用例详细设计**

**2.1 子程序文件上传功能测试**

**测试用例 TC001: 正常上传子程序文件测试**

**用例ID：**TC001  
**用例名称：**正常上传子程序文件测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**系统正常运行，文件存储目录有写入权限

**测试数据**

**请求方法：**POST  
**请求路径：**/file/upload/subprogram  
**Content-Type：**multipart/form-data  
**文件参数：**file (test\_subprogram.f90, 1KB)

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "文件上传成功", "data": "test\_subprogram\_20250807\_123456.f90" }

**测试用例 TC002: 上传空文件测试**

**用例ID：**TC002  
**用例名称：**上传空文件测试  
**测试类型：**异常测试  
**优先级：**中  
**前置条件：**系统正常运行

**测试数据**

**请求方法：**POST  
**请求路径：**/file/upload/subprogram  
**Content-Type：**multipart/form-data  
**文件参数：**file (empty\_file.txt, 0KB)

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "文件不能为空", "data": null }

**2.2 子程序文件下载功能测试**

**测试用例 TC003: 正常下载子程序文件测试**

**用例ID：**TC003  
**用例名称：**正常下载子程序文件测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**系统中存在要下载的文件

**测试数据**

**请求方法：**GET  
**请求路径：**/file/download/subprogram/test\_subprogram.f90  
**路径参数：**fileName = test\_subprogram.f90

**预期结果**

**状态码：**200  
**响应格式：**文件流  
**响应头：**Content-Disposition: attachment; filename=test\_subprogram.f90

**测试用例 TC004: 下载不存在的文件测试**

**用例ID：**TC004  
**用例名称：**下载不存在的文件测试  
**测试类型：**异常测试  
**优先级：**中  
**前置条件：**系统中不存在要下载的文件

**测试数据**

**请求方法：**GET  
**请求路径：**

/file/download/subprogram/nonexistent\_file.f90  
**路径参数：**fileName = nonexistent\_file.f90

**预期结果**

**状态码：**404  
**响应格式：**无内容  
**响应内容：**文件不存在

**2.3 仿真输入文件上传功能测试**

**测试用例 TC005: 正常上传仿真输入文件测试**

**用例ID：**TC005  
**用例名称：**正常上传仿真输入文件测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**系统正常运行，文件存储目录有写入权限

**测试数据**

**请求方法：**POST  
**请求路径：**/file/upload/simulation  
**Content-Type：**multipart/form-data  
**文件参数：**file (simulation\_input.dat, 2KB)

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "文件上传成功", "data": "simulation\_input\_20250807\_123456.dat" }

**测试用例 TC006: 上传超大文件测试**

**用例ID：**TC006  
**用例名称：**上传超大文件测试  
**测试类型：**异常测试  
**优先级：**中  
**前置条件：**系统正常运行

**测试数据**

**请求方法：**POST  
**请求路径：**/file/upload/simulation  
**Content-Type：**multipart/form-data  
**文件参数：**file (large\_file.dat, 100MB)

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "文件大小超出限制", "data": null }

**2.4 仿真输入文件下载功能测试**

**测试用例 TC007: 正常下载仿真输入文件测试**

**用例ID：**TC007  
**用例名称：**正常下载仿真输入文件测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**系统中存在要下载的文件

**测试数据**

**请求方法：**GET  
**请求路径：**/file/download/simulation/simulation\_input.dat  
**路径参数：**fileName = simulation\_input.dat

**预期结果**

**状态码：**200  
**响应格式：**文件流  
**响应头：**Content-Disposition: attachment; filename=simulation\_input.dat

**测试用例 TC008: 文件名包含特殊字符下载测试**

**用例ID：**TC008  
**用例名称：**文件名包含特殊字符下载测试  
**测试类型：**异常测试  
**优先级：**低  
**前置条件：**系统中存在包含特殊字符的文件名

**测试数据**

**请求方法：**GET  
**请求路径：**/file/download/simulation/test%20file%20with%20spaces.dat  
**路径参数：**fileName = test file with spaces.dat

**预期结果**

**状态码：**404  
**响应格式：**无内容  
**响应内容：**文件不存在或路径错误

**三、测试数据准备**

**3.1 测试文件准备**

-- 测试文件列表 1. test\_subprogram.f90 (1KB) - 子程序测试文件 2. simulation\_input.dat (2KB) - 仿真输入测试文件 3. empty\_file.txt (0KB) - 空文件 4. large\_file.dat (100MB) - 超大文件 5. test file with spaces.dat (1KB) - 包含空格的文件名

**3.2 测试环境变量**

// Apifox环境变量配置 { "baseUrl": "http://localhost:8081", "subprogramFileName": "test\_subprogram.f90", "simulationFileName": "simulation\_input.dat", "maxFileSize": "10MB" }

**3.3 文件存储路径配置**

-- 文件存储路径（需要在配置文件中设置） subprogram.upload.path: /data/files/subprogram/ simulation.upload.path: /data/files/simulation/ max.file.size: 10MB

**组织管理功能测试用例文档**

**文档信息**

**项目名称：**工业软件后端系统  
**功能模块：**组织管理功能  
**接口路径：**/organizations  
**请求方法：**GET/POST/DELETE  
**文档版本：**v1.0  
**创建日期：**2025-08-07  
**测试工具：**Apifox

**一、测试概述**

**1.1 测试目标**

验证组织管理功能的正确性、稳定性和安全性，包括组织分页查询、组织创建、成员管理等功能。

**1.2 测试环境**

**Base URL：**http://localhost:8081  
**数据库：**MySQL 8.0  
**缓存：**Redis  
**测试工具：**Apifox

**二、测试用例详细设计**

**2.1 组织分页查询功能测试**

**测试用例 TC001: 正常分页查询组织列表测试**

**用例ID：**TC001  
**用例名称：**正常分页查询组织列表测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**系统中存在组织数据

**测试数据**

**请求方法：**GET  
**请求路径：**/organizations  
**查询参数：**pageNum=1&pageSize=10&keyword=测试

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "操作成功", "data": { "records": [ { "id": 1, "orgName": "测试组织1", "description": "测试组织描述", "createTime": "2025-08-07 10:00:00", "memberCount": 5 } ], "total": 1, "pageNum": 1, "pageSize": 10 } }

**测试用例 TC002: 空关键词查询组织列表测试**

**用例ID：**TC002  
**用例名称：**空关键词查询组织列表测试  
**测试类型：**功能测试  
**优先级：**中  
**前置条件：**系统中存在组织数据

**测试数据**

**请求方法：**GET  
**请求路径：**/organizations  
**查询参数：**pageNum=1&pageSize=10&keyword=

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**返回所有组织列表

**2.2 组织创建功能测试**

**测试用例 TC003: 正常创建组织测试**

**用例ID：**TC003  
**用例名称：**正常创建组织测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**系统正常运行，用户已登录

**测试数据**

**请求方法：**POST  
**请求路径：**/organizations  
**Content-Type：**application/json  
**请求体：**

{ "orgName": "新测试组织", "description": "这是一个新的测试组织" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "组织创建成功", "data": null }

**测试用例 TC004: 创建重复名称组织测试**

**用例ID：**TC004  
**用例名称：**创建重复名称组织测试  
**测试类型：**异常测试  
**优先级：**中  
**前置条件：**系统中已存在同名组织

**测试数据**

**请求方法：**POST  
**请求路径：**/organizations  
**Content-Type：**application/json  
**请求体：**

{ "orgName": "已存在的组织", "description": "重复的组织名称" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "组织名称已存在", "data": null }

**2.3 组织成员管理功能测试**

**测试用例 TC005: 正常添加成员到组织测试**

**用例ID：**TC005  
**用例名称：**正常添加成员到组织测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**组织存在，用户存在且未加入该组织

**测试数据**

**请求方法：**POST  
**请求路径：**/organizations/1/members  
**Content-Type：**application/json  
**请求体：**

{ "userIds": [2, 3, 4] }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "成员添加成功", "data": null }

**测试用例 TC006: 添加已存在成员测试**

**用例ID：**TC006  
**用例名称：**添加已存在成员测试  
**测试类型：**异常测试  
**优先级：**中  
**前置条件：**用户已在该组织中

**测试数据**

**请求方法：**POST  
**请求路径：**/organizations/1/members  
**Content-Type：**application/json  
**请求体：**

{ "userIds": [1] }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "用户已在组织中", "data": null }

**测试用例 TC007: 正常移除组织成员测试**

**用例ID：**TC007  
**用例名称：**正常移除组织成员测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**用户在组织中

**测试数据**

**请求方法：**DELETE  
**请求路径：**/organizations/1/members/2  
**路径参数：**orgId=1, memberId=2

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "成员移除成功", "data": null }

**测试用例 TC008: 移除不存在的成员测试**

**用例ID：**TC008  
**用例名称：**移除不存在的成员测试  
**测试类型：**异常测试  
**优先级：**中  
**前置条件：**用户不在该组织中

**测试数据**

**请求方法：**DELETE  
**请求路径：**/organizations/1/members/999  
**路径参数：**orgId=1, memberId=999

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "用户不在该组织中", "data": null }

**2.4 组织成员查询功能测试**

**测试用例 TC009: 正常查询组织成员列表测试**

**用例ID：**TC009  
**用例名称：**正常查询组织成员列表测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**组织存在且有成员

**测试数据**

**请求方法：**GET  
**请求路径：**/organizations/1/members  
**查询参数：**pageNum=1&pageSize=10  
**路径参数：**orgId=1

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "操作成功", "data": { "records": [ { "id": 1, "username": "user1", "email": "user1@example.com", "joinTime": "2025-08-07 10:00:00" } ], "total": 1, "pageNum": 1, "pageSize": 10 } }

**测试用例 TC010: 查询不存在组织的成员测试**

**用例ID：**TC010  
**用例名称：**查询不存在组织的成员测试  
**测试类型：**异常测试  
**优先级：**中  
**前置条件：**组织不存在

**测试数据**

**请求方法：**GET  
**请求路径：**/organizations/999/members  
**查询参数：**pageNum=1&pageSize=10  
**路径参数：**orgId=999

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "组织不存在", "data": null }

**2.5 未分配成员查询功能测试**

**测试用例 TC011: 正常查询未分配成员列表测试**

**用例ID：**TC011  
**用例名称：**正常查询未分配成员列表测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**系统中存在未分配组织的用户

**测试数据**

**请求方法：**GET  
**请求路径：**/organizations/unassigned-members  
**查询参数：**pageNum=1&pageSize=10

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "操作成功", "data": { "records": [ { "id": 5, "username": "unassigned\_user", "email": "unassigned@example.com" } ], "total": 1, "pageNum": 1, "pageSize": 10 } }

**测试用例 TC012: 查询未分配成员空结果测试**

**用例ID：**TC012  
**用例名称：**查询未分配成员空结果测试  
**测试类型：**功能测试  
**优先级：**低  
**前置条件：**所有用户都已分配到组织

**测试数据**

**请求方法：**GET  
**请求路径：**/organizations/unassigned-members  
**查询参数：**pageNum=1&pageSize=10

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "操作成功", "data": { "records": [], "total": 0, "pageNum": 1, "pageSize": 10 } }

**任务管理功能测试用例文档**

**文档信息**

**项目名称：**工业软件后端系统  
**功能模块：**任务管理功能  
**接口路径：**/modTasks  
**请求方法：**GET/POST/DELETE  
**文档版本：**v1.0  
**创建日期：**2025-08-07  
**测试工具：**Apifox

**一、测试概述**

**1.1 测试目标**

验证任务管理功能的正确性、稳定性和安全性，包括共享任务分页查询、私人任务分页查询、任务创建、任务删除等功能。

**1.2 测试环境**

**Base URL：**http://localhost:8081  
**数据库：**MySQL 8.0  
**缓存：**Redis  
**测试工具：**Apifox

**二、测试用例详细设计**

**2.1 共享任务分页查询功能测试**

**测试用例 TC001: 正常分页查询共享任务列表测试**

**用例ID：**TC001  
**用例名称：**正常分页查询共享任务列表测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**项目中存在共享任务数据

**测试数据**

**请求方法：**POST  
**请求路径：**/modTasks/shared/1/page  
**Content-Type：**application/json  
**路径参数：**projectId=1  
**请求体：**

{ "pageNum": 1, "pageSize": 10 }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "操作成功", "data": { "records": [ { "id": 1, "taskName": "测试共享任务1", "description": "这是一个测试共享任务", "status": "进行中", "priority": "高", "createTime": "2025-08-07 10:00:00", "assignee": "张三" } ], "total": 1, "pageNum": 1, "pageSize": 10 } }

**测试用例 TC002: 查询不存在项目的共享任务测试**

**用例ID：**TC002  
**用例名称：**查询不存在项目的共享任务测试  
**测试类型：**异常测试  
**优先级：**中  
**前置条件：**项目不存在

**测试数据**

**请求方法：**POST  
**请求路径：**/modTasks/shared/999/page  
**Content-Type：**application/json  
**路径参数：**projectId=999  
**请求体：**

{ "pageNum": 1, "pageSize": 10 }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "项目不存在", "data": null }

**2.2 私人任务分页查询功能测试**

**测试用例 TC003: 正常分页查询私人任务列表测试**

**用例ID：**TC003  
**用例名称：**正常分页查询私人任务列表测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**项目中存在私人任务数据

**测试数据**

**请求方法：**POST  
**请求路径：**/modTasks/private/1/page  
**Content-Type：**application/json  
**路径参数：**projectId=1  
**请求体：**

{ "pageNum": 1, "pageSize": 10 }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "操作成功", "data": { "records": [ { "id": 2, "taskName": "测试私人任务1", "description": "这是一个测试私人任务", "status": "待开始", "priority": "中", "createTime": "2025-08-07 11:00:00", "assignee": "李四" } ], "total": 1, "pageNum": 1, "pageSize": 10 } }

**测试用例 TC004: 查询不存在项目的私人任务测试**

**用例ID：**TC004  
**用例名称：**查询不存在项目的私人任务测试  
**测试类型：**异常测试  
**优先级：**中  
**前置条件：**项目不存在

**测试数据**

**请求方法：**POST  
**请求路径：**/modTasks/private/999/page  
**Content-Type：**application/json  
**路径参数：**projectId=999  
**请求体：**

{ "pageNum": 1, "pageSize": 10 }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "项目不存在", "data": null }

**2.3 共享任务创建功能测试**

**测试用例 TC005: 正常创建共享任务测试**

**用例ID：**TC005  
**用例名称：**正常创建共享任务测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**项目存在，用户已登录

**测试数据**

**请求方法：**POST  
**请求路径：**/modTasks/shared/1/create  
**Content-Type：**application/json  
**路径参数：**projectId=1  
**请求体：**

{ "taskName": "新共享任务", "description": "这是一个新的共享任务", "priority": "高", "assignee": "张三", "deadline": "2025-08-15" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "共享任务创建成功", "data": null }

**测试用例 TC006: 创建重复名称共享任务测试**

**用例ID：**TC006  
**用例名称：**创建重复名称共享任务测试  
**测试类型：**异常测试  
**优先级：**中  
**前置条件：**项目中已存在同名任务

**测试数据**

**请求方法：**POST  
**请求路径：**/modTasks/shared/1/create  
**Content-Type：**application/json  
**路径参数：**projectId=1  
**请求体：**

{ "taskName": "测试共享任务1", "description": "重复的任务名称", "priority": "中", "assignee": "李四", "deadline": "2025-08-20" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "任务名称已存在", "data": null }

**2.4 私人任务创建功能测试**

**测试用例 TC007: 正常创建私人任务测试**

**用例ID：**TC007  
**用例名称：**正常创建私人任务测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**项目存在，用户已登录

**测试数据**

**请求方法：**POST  
**请求路径：**/modTasks/private/1/create  
**Content-Type：**application/json  
**路径参数：**projectId=1  
**请求体：**

{ "taskName": "新私人任务", "description": "这是一个新的私人任务", "priority": "低", "assignee": "王五", "deadline": "2025-08-25" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "私人任务创建成功", "data": null }

**测试用例 TC008: 创建重复名称私人任务测试**

**用例ID：**TC008  
**用例名称：**创建重复名称私人任务测试  
**测试类型：**异常测试  
**优先级：**中  
**前置条件：**项目中已存在同名任务

**测试数据**

**请求方法：**POST  
**请求路径：**/modTasks/private/1/create  
**Content-Type：**application/json  
**路径参数：**projectId=1  
**请求体：**

{ "taskName": "测试私人任务1", "description": "重复的任务名称", "priority": "中", "assignee": "赵六", "deadline": "2025-08-30" }

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "任务名称已存在", "data": null }

**2.5 任务删除功能测试**

**测试用例 TC009: 正常删除任务测试**

**用例ID：**TC009  
**用例名称：**正常删除任务测试  
**测试类型：**功能测试  
**优先级：**高  
**前置条件：**任务存在

**测试数据**

**请求方法：**DELETE  
**请求路径：**/modTasks/delete/1  
**路径参数：**taskId=1

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": 200, "message": "任务删除成功", "data": null }

**测试用例 TC010: 删除不存在任务测试**

**用例ID：**TC010  
**用例名称：**删除不存在任务测试  
**测试类型：**异常测试  
**优先级：**中  
**前置条件：**任务不存在

**测试数据**

**请求方法：**DELETE  
**请求路径：**/modTasks/delete/999  
**路径参数：**taskId=999

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "任务不存在", "data": null }

**测试用例 TC011: 删除已完成任务测试**

**用例ID：**TC011  
**用例名称：**删除已完成任务测试  
**测试类型：**异常测试  
**优先级：**低  
**前置条件：**任务已完成

**测试数据**

**请求方法：**DELETE  
**请求路径：**/modTasks/delete/3  
**路径参数：**taskId=3

**预期结果**

**状态码：**200  
**响应格式：**JSON  
**响应内容：**

{ "code": -1, "message": "已完成的任务不能删除", "data": null }