|  |  |  |
| --- | --- | --- |
| 文件状态：  [ ] 草稿  [ √ ] 正式发布  [] 正在修改 | 文件标识： | PSY-IT-01 |
| 当前版本： | V1.0 |
| 作 者： | 王晨楠 |
| 完成日期： | 2025-4-1 |

KomeijiHelper

接口测试计划

版 本 历 史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本/状态 | 作者 | 参与者 | 起止日期 | 备注 |
| V0.1 | 王晨楠 | 朱炎、陈俊伟 | 2025.3.18-2025.3.22 | 初版 |
| V0.1 | 王晨楠 | 朱炎、陈俊伟 | 2025.4.1- | 正式版 |
|  |  |  |  |  |

目录

[1 概述 3](#_Toc194344407)

[1.1 软件概述 3](#_Toc194344408)

[1.2 文档概述 3](#_Toc194344409)

[2 引用文件 3](#_Toc194344410)

[3 测试准备 3](#_Toc194344411)

[3.1 测试环境规划 3](#_Toc194344412)

[3.2 测试环境示意图 4](#_Toc194344413)

[3.3 测试策略 4](#_Toc194344414)

[4 测试说明 5](#_Toc194344415)

[4.1 测试项目描述 5](#_Toc194344416)

[4.2 测试用例 5](#_Toc194344417)

[5 测试实施安排 6](#_Toc194344418)

[5.1 测试进度 6](#_Toc194344419)

[5.2 缺陷管理 6](#_Toc194344420)

[6 接口测试规程 9](#_Toc194344421)

[6.1 启动标准 9](#_Toc194344422)

[6.2 中止标准 9](#_Toc194344423)

[6.3 通过标准 9](#_Toc194344424)

[7 测试风险分析 9](#_Toc194344425)

# 概述

## 软件概述

软件主要功能：见附件《g2\_komeijihelper接口文档》与《g2\_komeijihelper系统设计文档》

## 文档概述

本文档描述了线上心理咨询系统komeijihelper的软件接口测试，包括测试标准、测试策略、测试样例设计方法等。文档的第二部分介绍了引用的文件，第三部分给出了测试准备，测试策略在第四部分给出，第五部分给出了测试说明（包括用例设计），第六第七部分对测试的过程做出了安排，最后的在第八部分结合测试结果进行风险分析。

# 引用文件

《g2\_komeijihelper系统软件需求》

《g2\_komeijihelper项目技术可行性报告》

《g2\_komeijihelper业务分析说明书》

# 测试准备

## 测试环境规划

表 1 测试环境

|  |  |  |
| --- | --- | --- |
| **软/硬件** | **版本/配置** | **用途** |
| 硬件 | Intel i7, 16GB RAM, 512GB SSD | 工作机 |
| 硬件 | 8核CPU, 32GB RAM, 1TB SSD | 服务器 |
| 软件 | Windows11 | 工作机操作系统 |
| 软件 | Postman 11.37.1 | 接口测试工具 |
| 软件 | Ubuntu 20.04 LTS | 服务器操作系统 |
| 软件 | github | 接口持续集成及监控软件 |

## 测试环境示意图

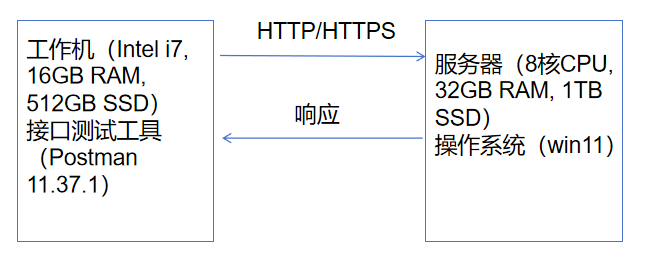


图 1 测试环境示意图

## 测试策略

本节描述了用于本次komeijihelper线上心理咨询系统接口测试的策略和方法，包括测试用例设计方法、测试数据选取原则、预期结果定义原则、实测结果获取与判定原则、测试脚本设计与实现原则以及自动化测试架构等内容。测试策略旨在确保接口功能的全面验证、性能的稳定性以及系统的可靠性。

测试用例设计采用以下方法：

**1.接口测试场景设计**：基于《g2\_komeijihelper系统软件需求》中的功能点，覆盖正常场景、边界场景及异常场景（如无效输入、超载请求）。

**2.测试数据选取原则**：包括合法数据（如正确用户名密码）、非法数据（如超长字符串、空值）、边界数据（如最大用户数）

**3.预期结果定义原则**：依据接口设计文档，定义每个接口的返回状态码、响应数据结构及错误信息。

**4.实测结果获取/判定原则**：通过Postman记录实际响应，与预期结果对比，判定是否通过。

**5.测试脚本设计和实现原则**：使用Postman脚本实现自动化测试，结合GitHub Actions进行持续集成。

**6.自动化测试架构**：基于Postman集合+Newman命令行工具，集成至GitHub Actions，实现接口自动化回归测试。

# 测试说明

## 测试项目描述

表 2 测试项目描述

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **测试项名称** | **测试项标识** | **测试子项标识** | **说明** |
| *1* | *用户相关API* | *PSY\_User\_API* | PSY\_User\_API\_login | 验证用户后台登录接口实现是否满足接口设计要求 |
| *2* | *用户相关API* | *PSY\_User\_API* | PSY\_User\_API\_logout | 验证用户后台注销接口实现是否满足接口设计要求 |
| *3* | *用户相关API* | *PSY\_User\_API* | PSY\_User\_API\_getUsersByClass | 验证根据用户名获取用户类型实现是否满足接口设计要求 |
| *4* | *用户相关API* | *PSY\_User\_API* | *PSY\_User\_API\_register* | 验证用户注册接口实现是否满足接口设计要求 |
| *5* | *用户相关API* | *PSY\_User\_API* | *PSY\_User\_API\_getUserName* | 验证获取用户名是否满足接口设计要求 |
| *7* | *会话API* | *PSY\_Session\_API* | *PSY\_Session\_API\_Start* | 验证发起会话是否满足接口设计要求 |
| *8* | *会话API* | *PSY\_Session\_API* | *PSY\_Session\_API\_End* | 验证结束会话是否满足接口设计要求 |

## 测试用例

参见《komeijihelper接口测试用例.xlsx》

# 测试实施安排

## 测试进度

表 3 测试项目

|  |  |  |  |
| --- | --- | --- | --- |
| **任务名称** | **责任人** | **任务起止时间** | **输出产物** |
| 测试用户相关接口 | 王晨楠、陈俊伟 | 2025.3.19-2025.4.2 | 用户相关接口测试报告 |
| 测试会话接口 | 王晨楠、李智、陈俊桥 | 2025.3.19-2025.4.2 | 会话接口测试报告 |
| 接口安全测试 | 王晨楠 | 2025.4.6-2025.4.8 | 接口安全测试报告 |

## 缺陷管理

5.2.1 缺陷管理流程

在本次线上心理咨询系统测试中，缺陷管理将通过版本库服务器（如 Gitee 或 GitHub）进行统一记录和管理。具体流程如下：

1. 缺陷提交：

测试人员在测试过程中发现缺陷后，在 Gitee/GitHub 等版本库服务器中提交 Issue，选择类型为“缺陷”（Bug）。

提交时需详细描述缺陷现象、复现步骤、预期结果与实际结果，并附上相关截图、日志或接口请求/响应数据。

建议将缺陷 Issue 提交给项目经理。

2. 缺陷审核：

测试经理收到缺陷后进行初步审核，判断是否为真实缺陷，并评估其严重程度。

若确认为缺陷，经理根据缺陷涉及的接口或功能模块，优先分配给相关开发人员处理。

3. 缺陷修复：

开发人员根据分配的 Issue 进行缺陷修复，修复完成后在版本库中提交代码，并更新 Issue 状态为“待验证”，注明修复说明。

4. 缺陷验证：

缺陷提交人（即发现缺陷的测试人员）对修复结果进行验证，检查是否符合预期。

若验证通过，关闭 Issue；若未通过，重新标记为“开放”并补充说明，退回开发人员继续处理。

5. 缺陷跟踪：

所有缺陷的状态和处理进展将在版本库中实时更新，确保缺陷从提交到关闭的全程可追溯。

5.2.2 缺陷严重程度等级说明

为确保缺陷处理的优先级和效率，根据线上心理咨询系统的业务特性和测试目标，将缺陷分为以下四个严重程度等级：

1. 致命性缺陷（Critical）：

定义：系统核心功能无法使用，或错误实现关键接口，导致服务不可用；测试过程中出现系统自动退出、死机、崩溃，或对用户数据（如用户信息、会话记录）造成破坏。

示例：

- 用户登录接口（/user/login）返回错误，导致所有用户无法登录。

- 会话创建接口（/session/create）异常中断，导致咨询服务无法正常进行。

- 数据丢失或泄露，如用户个人信息被错误覆盖。

- 处理优先级：最高，需立即修复并验证。

2. 严重性缺陷（High：）

- 定义：接口基本功能未实现且无替代方案；重要接口部分功能缺失且无替代方法；性能未达标（如响应时间超过预期 500ms）。

- 示例：

- 微信小程序登录接口（/visitor/vx/login）无法返回 token，用户无法登录。

- 获取当前用户信息接口（/user/me）缺少关键字段（如 roleId），影响权限判断。

- 高并发下接口响应时间超标，影响用户体验。

- 处理优先级：高，需尽快修复。

3. 一般性缺陷（Medium）：

- 定义：与严重性缺陷类似，但存在替代方案；对误操作或错误输入缺乏提示，导致用户体验下降但不影响核心功能。

- 示例：

- 添加督导用户接口（/user/supervisor/add）未校验用户名重复，但可通过手动检查避免。

- 用户注销接口（/user/logout）未提示“是否确认注销”，但功能正常。

- 统计接口（/statistic/recent）数据格式不一致，但不影响数据读取。

- 处理优先级：中等，可在迭代中修复。

4. 建议性缺陷（Low）：

- 定义：易用性或用户体验相关问题，如提示信息不友好、界面风格不一致、中英文混杂等，不影响功能实现。

- 示例：

- 登录失败时返回“Bad credentials”，未翻译为中文“用户名或密码错误”。

- 界面按钮样式在不同页面不统一。

- 响应消息中“操作成功”与“Success”混用。

- 处理优先级：低，可根据开发资源安排优化。

5.2.3 缺陷管理工具与记录要求

- 工具：使用 Gitee/GitHub 的 Issue 管理功能，结合 Postman 的测试结果导出（如 JSON 或 HTML 格式）作为附件。

- 记录要求：

- 缺陷标题：简要描述问题，如“[致命] 用户登录接口崩溃”。

- 缺陷描述：包括复现步骤、环境信息（如接口地址、请求头、请求体）、预期结果和实际结果。

- 附件：测试日志、接口响应截图等。

- 表情：按严重程度添加标签（如 Critical、High、Medium、Low），便于分类和统计。

- 统计与报告：测试结束后，生成缺陷统计报告，包括缺陷总数、各严重程度分布、修复率等，提交给项目经理和开发团队。

5.2.4 注意事项

- 及时性：致命性和严重性缺陷需在发现后 24 小时内分配并开始修复。

- 沟通：若缺陷涉及多个模块或责任不明，测试经理需组织相关人员讨论确认。

- 回归测试：修复后的缺陷需进行回归测试，确保未引入新问题。

- 数据保护：涉及用户敏感信息（如手机号、姓名）的缺陷描述需脱敏处理，避免泄露。

# 接口测试规程

## 启动标准

1. 测试环境满足计划需求，人员到位
2. 接口测试用例设计完成且通过评审

## 中止标准

1. 测试环境或关键系统不可用
2. 出现宕机、不响应等致命的系统失效

## 通过标准

1. 所有接口测试用例执行完毕
2. 确认的提交缺陷已修改完毕

# 测试风险分析

表 4 测试风险分析

|  |  |  |  |
| --- | --- | --- | --- |
| **风险标识** | **风险描述** | **危害程度（H/M/L）** | **对策** |
| R01 | 测试环境搭建延迟 | M | 提前准备备用硬件资源，制定环境搭建检查清单 |
| R02 | 接口实现与需求不符 | H | 在测试前与开发团队确认接口设计一致性 |
| R03 | 测试数据不足 | M | 提前准备多样化测试数据，覆盖多种场景 |
| R04 | 缺陷修复不及时 | H | 设置缺陷优先级，安排额外开发资源支持 |
| R05 | 自动化测试脚本失效 | L | 定期检查并更新脚本，确保与接口变更同步 |