项目位置：[https://github.com/NINEMINEsigma/202412-LLM-](https://github.com/NINEMINEsigma/202412-LLM-.git)

Lekit源码：<https://github.com/NINEMINEsigma/lekit>

**目 录**

[第1章 作品概述 1](#_Toc185375029)

[1.1 创意来源 1](#_Toc185375030)

[1.2 产生背景 1](#_Toc185375031)

[1.3 用户群体 1](#_Toc185375032)

[1.4 主要功能与特色 1](#_Toc185375033)

[1.5 应用价值 1](#_Toc185375034)

[1.6 推广前景 2](#_Toc185375035)

[第2章 问题分析 2](#_Toc185375036)

[2.1 问题来源 2](#_Toc185375037)

[2.2 现有解决方案 2](#_Toc185375038)

[2.3 本作品要解决的痛点问题 3](#_Toc185375039)

[2.4 解决问题的思路 3](#_Toc185375040)

[2.4.1 功能模块 3](#_Toc185375041)

[2.4.2 性能需求 3](#_Toc185375042)

[2.4.3 项目自测 3](#_Toc185375043)

[第3章 技术方案 4](#_Toc185375044)

[3.1 技术路线框架 4](#_Toc185375045)

[3.2 XX模块 4](#_Toc185375046)

[3.2.1 XX 4](#_Toc185375047)

[3.2.2 XX 5](#_Toc185375048)

[3.3 XX模块 5](#_Toc185375049)

[3.3.1 XX 5](#_Toc185375050)

[3.3.2 XX 5](#_Toc185375051)

[3.4 XX模块 5](#_Toc185375052)

[3.4.1 XX 5](#_Toc185375053)

[3.5 XX模块 5](#_Toc185375054)

[3.5.1 XX 5](#_Toc185375055)

[3.5.2 XX 5](#_Toc185375056)

[第4章 系统实现 6](#_Toc185375057)

[4.1 软件设计实现 6](#_Toc185375058)

[4.1.1 前端技术 6](#_Toc185375059)

[4.1.2 后端技术 6](#_Toc185375060)

[4.1.3 接口封装与调用 6](#_Toc185375061)

[4.1.4 系统集成 6](#_Toc185375062)

[4.2 用户界面 6](#_Toc185375063)

[4.2.1 主要使用界面 6](#_Toc185375064)

[4.3 数据来源 6](#_Toc185375065)

[4.4 数据训练 6](#_Toc185375066)

[4.5 改进过程 7](#_Toc185375067)

[4.5.1 XX 7](#_Toc185375068)

[4.5.2 XX系列模型的选择 8](#_Toc185375069)

[4.6 系统部署方法 8](#_Toc185375070)

[第5章 测试分析 9](#_Toc185375071)

[5.1.1 XX模块测试分析 9](#_Toc185375072)

[5.1.2 XX模块测试分析 9](#_Toc185375073)

[第6章 作品总结 9](#_Toc185375074)

[6.1 作品特色与创新点 9](#_Toc185375075)

[6.1.1 作品特色 9](#_Toc185375076)

[6.1.2 作品创新点 9](#_Toc185375077)

[6.2 应用推广 10](#_Toc185375078)

[6.3 作品展望 10](#_Toc185375079)

[参考文献 10](#_Toc185375080)

# 作品概述

## 创意来源

暂空。

## 产生背景

暂空。

## 用户群体

暂空。

## 主要功能与特色

主要功能：

暂空。

//可插图

特色：

1. **XXX：**暂空。

## 应用价值

简述暂空。

1. **XXX：**暂空。

## 推广前景

暂空。

# 问题分析

## 问题来源

暂空。

## 现有解决方案

暂空。

表 X XXX工具/产品/方法调研

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **工具/产品/方法** | **效率** | **成本** | **XX能力** | **XX内容** | **稳定性** | **生成模式** |
| 传统创作 | 低 | 高 | 极高 | 连续 | 无 | 固定 |
| XXX | 高 | 低 | 中 | 不连续 | 稳定 | 固定 |
| XXX | 高 | 低 | 中 | 连续 | 稳定 | 固定 |
| XXX | 高 | 低 | 中 | 连续 | 稳定 | 固定 |
| MyProject | 中 | 低 | 高 | 连续 | 较稳定 | 可定制 |

//可插图

**与XX相比本平台的优势：**

1. **效率与速度：**暂空。
2. **一致性与多样性：**暂空。
3. **节省成本：**暂空。

**与同类AI工具相比本平台的优势：**

1. **XX：**暂空。
2. **XX：**暂空。

**平台有待优化之处：**

1. **相较于同类AI工具XX的不足：**暂空。
2. **系统维护的稳定性：**暂空。

## 本作品要解决的痛点问题

1. 暂空。
2. 暂空。

## 解决问题的思路

### 功能模块

1. **低层工具模块：**lekit库（lazy and easy kit），提供了项目中使用的绝大多数接口与工具，在使得代码更加易读且明确的情况下，还推动了项目耦合性的大幅降低与提高了极大程度的可维护以及扩展性。经由lekit中的LangChain等重封装的核心，允许灵活地部署智能体并使用其服务，并通过定制的prompt及其重封装类的模板构造合适的上下文。
2. **用例生成模块：**大模型将通过阅读。此模块将生成可供人员阅读的测试用例文档以及后续测试工作的自动路径AutoRuntimePath配置（其中包含文档中的关键词与网址，以及可能的各种操作）。
3. **自动化测试模块：**依据用例生成模块的产物，在目标平台上执行自动化测试。进一步结合智能体与提供的工具函数，自动化测试将在智能体对设计文档的理解下逐步对各个用例进行测试并通过网页反馈、视觉表观等方面得到与预期结果是否具有差异，并得出该用例的评测结果。
4. **XX模块：**

### 性能需求

1. **XX模块：**暂空。
2. **XX模块：**暂空。

### 项目自测

* 未定

1. 数据格式：未定
2. 数据来源：未定
3. 数据获取方式：暂空
4. 数据特点：暂空
5. 数据规模：未定
6. 数据样例：

//可插图

1. 数据集用途：暂空

# 技术方案

## 技术路线框架

本项目的技术方案由高低两层技术架构组成。

第三方与底层API通过lekit部分聚合成多个核心，这些核心组成的低层架构将以更详细和统一的形式向高层技术架构提供具体实现而不需要考虑参数可能的类型与过程中存在的操作。

多个模块组成了高层技术架构，在这一层中的所有跨模块通信仅通过ProjectConfig类完成，所有的配置变量也由此类完成数据驱动，实现了代码中的低内聚，并且可以通过Config.json热更新而不需要停机

图 X 整体技术路线框架图

技术方案的具体层次结构和逻辑流向如下：

首先，暂空。

接着，暂空。

随后，暂空。

最后，暂空。

总结暂空。

## XX模块

：

图 5  文本理解模块技术架构图

### XX

暂空。

### XX

暂空。

## XX模块

暂空。

### XX

暂空。

### XX

暂空。

## XX模块

暂空。

### XX

暂空。

## XX模块

暂空。

### XX

暂空。

### XX

暂空。

# 系统实现

## 软件设计实现

### 前端技术

暂空。

### 后端技术

暂空。

### 接口封装与调用

暂空。

### 系统集成

暂空。

## 用户界面

### 主要使用界面

暂空。

## 数据来源

暂空。

## 数据训练

为了全面、系统地评估项目的性能与效果，特制定以下训练测试方案：

1. 训练目标
2. **功能完整性：**暂空。
3. **适应性与稳定性：**暂空。
4. **用户体验：**暂空。
5. 训练方法与步骤
6. 单元测试

简述暂空。

评估指标如下：

1. 文本理解准确率：统计正确文本内容所占的比例。
2. 生成时间：记录单个产出的平均耗时。
3. 结果评分：根据人工对照进行主观评分。
4. 系统测试

简述暂空。

评估指标如下：

1. 系统稳定性：记录处理过程中出现错误、延迟或其他异常的次数与比例。
2. XX：未定。
3. 数据分析
4. **数据整理：**汇总单元测试、系统测试与用户体验测试的各项量化指标及主观评价。
5. **性能分析：**对比测试结果与预期目标，分析项目在功能完整性、艺术表现力、适应性与稳定性、用户体验方面的实际表现，识别优势与不足。
6. **结果解读：**结合具体测试案例，深入解读数据背后的原因。
7. **建议提出：**根据测试结果与分析，为智绘连环平台的优化升级提供针对性改进建议，包括代码改进、参数调整等。

通过以上训练测试，在开发过程中定期对项目的综合性能进行全面、深入的评估，为其持续优化提供坚实的数据支持。

## 改进过程

### XX

简述暂空：

1. **实验设计：**暂空。
2. **数据收集：**暂空。
3. **数据处理：**暂空。
4. 测试结果：

暂空（XX从XX变为了XX…）。

### XX系列模型的选择

暂空。

## 系统部署方法

1. 系统需求
2. 硬件需求：

未定。

1. 网络环境：

需要网络 / 不需要网络。

1. 环境准备
2. 未定。
3. 暂空。
4. 部署

部署步骤：

1. 对项目的相关代码进行编译
2. 将服务端上传至服务器，并在指定或需求的目录下解压缩。
3. 部署XX模型到服务器机器的本地。
4. 根据首次启动的配置引导完成Config.json的配置
5. 服务启动与验证
6. 启动与测试：暂空。
7. 功能完整性验证：暂空。

# 测试分析

### XX模块测试分析

* 测试数据集

暂空。

* 测试过程

1. 数据预处理：暂空。

输出评估：未定。

* 测试结果

暂空。

* 分析与结论

暂空。

### XX模块测试分析

* 测试数据集

暂空。

# 作品总结

## 作品特色与创新点

### 作品特色

1. **XX：**暂空。
2. **XX：**暂空

### 作品创新点

1. **XX：**暂空

## 应用推广

1. **XX领域：**暂空

## 作品展望

//可插宣传图

暂空。

# 参考文献

[1]何文睿,高丹阳,周羿旭,等.基于扩散模型的多模态引导图像合成系统[J].北京信息科技大学学报(自然科学版),2023,38(06):80-87.

[2]刘泽润,尹宇飞,薛文灏,等.基于扩散模型的条件引导图像生成综述[J].浙江大学学报(理学版),2023,50(06):651-667.

//更多引用