

项目内容简介

项目开发背景

- 随着人工智能技术的飞速发展，AI 绘画技术也日趋成熟。近些日子来，无数精美的 AI 绘画作品都让我们眼前一新。对于没有接触过 AI 绘画领域的小白，他们可能也想体验 AI 绘画的奇妙，但不知从何下手。对于钻研 AI 绘画领域的技术人员，他们可能想分享自己的训练成果，同时和其他从业人员沟通交流，但是缺乏相关的平台。本软件的创建便是为了解决以上问题，为小白和技术人员提供一个在线生成图片，上传分享训练参数并和他人沟通的平台。

项目核心功能需求

- ① 用户能够在线欣赏、生成图片，下载相关模型参数
- ② 用户能够上传分享自己训练的模型参数
- ③ 用户能够使用软件系统提供的相关社交功能，与其他用户进行交流

本次里程碑任务

- 使用基于 Gradio + FastAPI 的 Web 应用快速开发部署技术，以 Restful 接口文档为功能开发核心导向，重构项目 AI 绘图生成服务的底层代码架构与算法逻辑。采用完全自主实现的轻量级任务队列 + 悲观锁机制，实现 Web 系统高并发情景下的 AI 绘图服务高可用。
- 在迭代 1 实现的系统功能上继续增量开发，优化已有功能的运行性能和 Web 界面样式，同时增量地实现了画廊作品展示、用户反馈、img2img、快速检索等一系列功能需求
- 配置华为云提供的 ECS 弹性云服务器，确定项目软件系统的部署图，在弹性云服务器上部署项目软件系统，实现了项目软件系统的部署与运行

本次里程碑任务

- 根据甲方的 SRS 需求文档，为软件系统增添了后台管理系统，实现了对用户账号、模型、反馈等相关数据的管理（将简单演示）
- 完善了基于 Apifox 的接口管理规范、测试用例和 Mock 数据，实现更加健全完善的项目自动化测试体系和管理流程。

工作项总览与相关图表

详见华为云



项目技术栈

- 主要开发语言：Python
- HTTP 服务器：Tornado + FastAPI
- 前端技术选型：Vue.js + Element UI/HTML + CSS + Js/Gradio
- 持久层框架：SQLAlchemy
- 数据库服务：MySQL + Redis
- 版本管理工具：Git
- 远程代码托管平台：华为云
- 接口管理与自动化测试工具：Apifox + Mock.js

目标和预期

- 围绕甲方 SRS 需求文档为核心，高质量地完成相关功能需求的开发任务
- 围绕功能用例，时刻开展自动化测试，保证项目的稳定性和可用性
- 上线项目，邀请用户体验并给出测试反馈，收集用户意见和 Bug 反馈，为第三次迭代的内容规划提供参考

信息和工具

- 小组中对于相关功能需求分析实现，很大程度上参考了甲方提供的 SRS 需求文档，以及甲方提供的用例图，从而使得设计与实现内容与甲方需求最大程度地做到匹配
- 技术上，我们小组借助相关网络资料以及以往的开发经验，并且和相关从业专业人士进行沟通，充分对于当前行业的流行技术生态进行调研，选择最有利于敏捷开发的技术栈，从而使得项目开发效率得到了很大的提升

困难与阻碍

- 我们小组开发在迭代 2 遇到的最大的困难是在项目部署到服务器运行并测试时，遇到项目在实际的网络环境与本地环境下运行的差异，导致项目之前的许多功能实现可能在本地运行高效、良好，但是在网络环境下则由于网络带宽、高并发场景等原因，导致项目运行效率低下。这导致我们虽然很快地实现了相关的项目需求，但是依然需要负责相关功能模块的同学花费许多的时间对于原本功能实现的底层算法与代码逻辑进行优化，尽可能地降低算法对于系统资源的消耗。这也使得我们深刻意识到了实际生产环境下的项目开发与测试的重要性，以及对于项目的高效性、可用性的重视程度，让我们对于实际可用的工业化项目的敏捷开发流程有了更加深刻的认识。

优势和创新

- 得益于我们小组在迭代 1 中搭建的良好项目管理架构，我们在迭代 2 中进一步对于整个管理流程进行了完善，高度规范化了组员从功能开发、自动化测试、代码提交、审查、合并、部署、上线等整个项目开发流程，从而使得我们小组的开发人员能够更加专注于项目的功能实现，而不是处理各种由于项目流程管理不当产生的问题，使得我们小组的开发效率得到了进一步的提升。
- 我们小组在第一次迭代的基础上，继续完善技术栈，加入了 Gradio + FastAPI 的技术选型实现 AI 画图服务，使得我们小组的项目更加完善，功能更加丰富，同时也使得我们小组的项目更加具有创新性。

结果和进度

- 我们小组在每周的组会上，都会各自汇报本周自主开发以及集中开发中完成任务的情况，并且由会议记录人员进行统计。最终结果表明，小组在本次迭代中，完成了甲方提出的所有功能需求，并且在自动化测试方面也取得了很大的进展，使得我们小组的项目更加稳定可靠，结果还是比较让我们感到欣喜和自豪的。
- 我们小组的开发进度推进稳健，在下一次迭代（第三次迭代）中，主要将会根据当前的相关 Bug 反馈和用户需求，对于系统的一些相关问题进行修复和优化。

情绪状态

- 我们小组成员在迭代开发中，由于本身都还有其它大量的学业和工作压力，因此在迭代开发过程中，有时候会出现一些情绪波动，影响打出代码的可靠性和质量，导致提交时审查门禁不通过，被发回修改。但是我们小组成员都能够很好地调整自己的情绪，保持积极向上的心态，最终完成了本次迭代的开发任务。

心得体会与开发意义

- 通过本次迭代任务，我们小组成员之间的配合更加融洽并且具有默契，能够作为一个团队更加高效与良好地完成相关的项目开发任务，并且更加熟悉了类似的大型项目的开发流程与管理模式，为之后的下一次迭代乃至今后学习工作中的类似项目开发情景打下了良好的基础。