



第十四届全国大学生软件创新大赛

文档编号: SWC2021- T20210267-搞快点



我的动态

iMoments

作品创新性分析报告

Version: [V2.3.1]



搞快点

2021-4-10

All Rights Reserved

目录

1	项目概述	1
1.1	项目背景	1
1.2	项目定位	1
1.2.1	应用场景	1
1.2.2	目标人群	1
1.3	项目方案	2
1.4	项目目标	2
1.5	项目价值	2
2	开发计划	3
2.1	最终呈现形式	3
2.2	主要功能描述	3
2.3	运行环境	3
2.4	验收标准	4
2.5	关键问题	4
2.6	进度安排	4
2.7	开发预算	4
3	可行性分析	5
3.1	技术可行性分析	5
3.2	资源可行性分析	5
3.3	市场可行性分析	5
4	需求分析	7
4.1	数据需求	7
4.1.1	静态数据	7
4.1.2	动态数据	7
4.1.3	数据词典	7
4.1.4	数据采集	7
4.2	功能需求	7
4.2.1	**功能模块	7
4.3	性能需求	10
4.3.1	时间特性	10
4.3.2	适应性	10
4.4	界面需求	11
4.5	接口需求	12

4.5.1	硬件接口	12
4.5.2	软件接口	12
4.6	其他需求.....	12
5	概要设计	13
5.1	处理流程.....	13
5.2	总体结构设计	14
5.3	功能设计.....	15
5.4	用户界面设计	15
5.5	数据结构设计	18
5.6	接口设计.....	18
5.6.1	外部接口	18
5.6.2	内部接口	19
5.7	错误/异常处理设计.....	20
5.7.1	错误/异常输出信息	20
5.7.2	错误/异常处理对策	21
5.8	系统配置策略.....	22
5.9	系统部署方案.....	23
5.10	其他相关技术与方案	23
6	数据库设计	24
7	手机端侧部署设计	26
7.1	手机环境需求.....	26
7.2	调用 AI Unit 能力接口设计（可选）	26
7.3	AI Boost 端侧模型加速设计（可选）	26
8	详细设计	27
8.1	文本预处理功能模块.....	27
8.1.1	功能描述	27
8.1.2	性能描述	27
8.1.3	输入	27
8.1.4	输出	27
8.1.5	程序逻辑	27
8.1.6	限制条件	27
8.2	图像预处理功能模块.....	27
8.2.1	功能描述	27
8.2.2	性能描述	28
8.2.3	输入	28
8.2.4	输出	28
8.2.5	程序逻辑	28

8.2.6	限制条件	28
8.3	素材检索与生成功能模块.....	28
8.3.1	功能描述	28
8.3.2	性能描述	28
8.3.3	输入	29
8.3.4	输出	29
8.3.5	程序逻辑	29
8.3.6	限制条件	29
8.4	账号信息功能模块.....	29
8.4.1	功能描述	29
8.4.2	性能描述	29
8.4.3	输入	29
8.4.4	输出	29
8.4.5	程序逻辑	29
8.4.6	限制条件	30

文档修订历史

序号	修订原因	版本号	作者	修订日期	备注
1	添加项目概述	V1.0.0	队员 D	2021-1-8	
2	添加可行性分析	V1.1.0	队员 B	2021-1-9	
3	添加数据需求	V1.2.0	队员 B	2021-1-9	
4	完成需求分析	V1.3.0	队员 A	2021-1-10	界面需求没写
5	修改需求分析	V1.3.1	队员 A	2021-1-10	改了一点点
6	添加开发计划	V1.4.0	队员 B	2021-1-10	
7	全文修改	V1.4.1	队员 B	2021-1-10	
8	添加界面需求	V1.4.2	队员 A	2021-1-10	
9	封面修改	V1.5.0	队员 D	2021-1-10	
10	最终审查修改	V1.5.1	队员 B	2021-1-10	
11	添加数据库设计	V2.0.0	队员 A	2021-4-7	以下为复赛
12	修改数据库设计	V2.0.1	队员 A	2021-4-8	
13	添加概要设计	V2.1.0	队员 D	2021-4-8	
14	添加详细设计	V2.1.1	队员 D	2021-4-8	
15	整体修改	V2.1.2	队员 A	2021-4-9	
16	更换图片 logo	V2.2.0	队员 A	2021-4-9	
17	完善概要设计	V2.2.1	队员 A	2021-4-10	
18	添加附件	V2.2.2	队员 A	2021-4-10	
19	完善概要设计	V2.3.0	队员 B	2021-4-10	
20	最近完善	V2.3.1	队员 B	2021-4-10	

1 项目概述

1.1 项目背景

随着移动互联网时代的全面来临，微信朋友圈、QQ 空间、微博等已经成为绝大多数国人记录生活分享快乐的聚集地。人们常常会通过文字加配图形式的动态，去向亲朋好友们展现自己的所见所闻和喜怒哀乐。

然而，你是否也曾遇到过这样的情况？手中只有 7 张照片想发朋友圈，却难受地发现凑不齐九宫格，有强迫症的你不得不放弃发出这条朋友圈。或者你已经上传了许多照片，却一时脑子短路，想不出该配些什么文字。更有甚者，你只是想表达你现在情绪，却发现无从下笔，只能干巴巴地写下几个字。这些都是不经意间发生在我们身上的事。

因此，有效解决这类“发动态困难症”问题，能够大幅提高人们网络生活幸福指数。

1.2 项目定位

1.2.1 应用场景

iMoments 是一款基于深度学习和端云结合技术的动态生成软件，旨在利用“端”上的个性化需求和“云”上的大量数据和强大的处理能力，帮助和引导用户完成一条动态的生成，解决用户配文难、补图难的问题，个性化定制动态的风格，让丰富多彩的动态变得触手可及。

其主要应用场景如下：

1. 个性化动态配文：

iMoments 操作简单清晰，用户可以轻松掌握基本的使用技巧，根据自身需求一键上传图片或文字，并根据提示的主题，生成有美感有意境的配文。

2. 智能凑图补图：

iMoments 结合云端，根据用户上传的图片和文字的分析结果，在云端海量图片库中进行查找，并根据用户特定需求来匹配相似的图片或相同含义的表情包，最后返回到用户端供用户挑选。并可以根据用户需要进行图片修复和简单滤镜修图。可以解决用户自身图片较少，但想要更多图片或表情包来发动态的需求。

1.2.2 目标人群

1. 有发动态需求的人。

2. 需要补图或简单配文的文章撰写、公众号运营的小编。

1.3 项目方案

iMoments 的功能实现可以分解为以下几个子问题及实现途径:

- ① 在用户上传的图片中提取关键要素: 百度云端通用物体和场景识别 API。
- ② 提取用户输入文字中的关键主题和情感倾向: 百度云端语义和情感倾向分析 API。
- ③ 生成配文: 本地 + 云端神经网络模型迭代求解。
- ④ 查找相似图片和相关表情包: 百度图像搜索 API。
- ⑤ 修复低像素和模糊图片: OPPO AI Unit 照片修复 API。
- ⑥ 简单的图片编辑功能 (套滤镜、调色调等): 本地图片处理。

备注:

由于项目使用的是 flutter 框架, oppo ai unit 也没有 flutter 支持, 所以很遗憾之前设想的照片降噪修复等功能无法实现。

相关图片的查找暂时只做了已有图片的浏览, 后续会加上对图片搜索 API 的调用。

1.4 项目目标

实现精准配文配图, 同时满足用户个性化需求, 高效短时地为用户提供动态发布的完美解决方案。

1.5 项目价值

1. 帮助大多数人完善动态发文, 高效提升朋友圈质量。
2. 节约自己撰写文字和寻找配图的时间, 更快速地解放发动态困难症。
3. 提供适当动态参考样例, 为官微动态提供解决方案。

2 开发计划

2.1 最终呈现形式

iMoments 最终呈现形式主要由手机端和云端两个部分组成。

手机端为一个 app，可以进行文字生成等负载较小的神经网络模型迭代。手机端需要调用云端服务器的 API 以实现图像要素提取、搜索类似图片和相关表情包等中间环节功能。

手机端需要通过云端协同工作，让云端分担本地的一部分性能开销（如照片超分辨率修复和通用物体识别等），也需要云端的大量图片库和强大的检索能力来得到相关的图片和表情。

2.2 主要功能描述

在首次启动手机端 iMoments 时，软件会给出配文的样例，用户可以选择喜欢的某些内容，iMoments 可以根据用户的选择明白用户的风格偏好。

用户使用软件时可以选择上传想发的图片或一段文字，并添加若干个主题关键字。iMoments 可以向云端发送请求，提取出图片的关键要素，分析文字的主题和情感倾向。根据分析图片和文字得到的信息，云端会在互联网和数据库中查找相关图片和表情包。最终云端会将图片和文字分析结果和相关图片和表情包的检索结果反馈给手机端。

在手机端，iMoments 可以通过云端反馈的关键词、情感倾向等信息，结合用户的风格偏好，利用事先训练的神经网络模型迭代生成有意境有美感的配文（或诗词等）。

iMoments 会将配文反馈给用户，并将相关的图片和表情包推荐给用户供用户自由选择。如果 iMoments 检测到用户的图片中有模糊或者分辨率低的图片，则会询问用户是否要在云端进行图片修复。

最终，用户还可以对配文进行编辑或选择将生成的配文复制到剪切板。并可以选择图片进行修图或直接保存到相册。修图时 iMoments 也会根据图片的风格和用户的情感倾向提供滤镜和色调调整等。

用户也可以选择收藏配文、图片、表情包等处理结果，并在收藏夹中查看。

2.3 运行环境

前端：安卓或 ios 平台 app，具有一定程度计算能力的手机。

后端：主流 Linux 或 WindowsServer 服务器平台。

2.4 验收标准

应用生成的配文语言流畅，具有一定的美感。

应用提供给用户的补图与主题相关，表情包符合用户想表达的情感。

应用运行稳定流畅，用户等待处理结果的时间在可接受范围内。

应用界面有一定设计感，符合用户审美。

2.5 关键问题

iMoments 需要生成与主题有关的有意境有美感的配文，并且能较为准确地表达出用户的感受。这就需要结合云端反馈的图片和文字要素相关性得分、情感倾向得分，以及用户提供的主题、用户的风格喜好，综合考虑并生成配文，并且神经网络模型也需要更复杂的训练。

2.6 进度安排

2-20 前完成软件整体架构设计，如前端界面和后端各类各函数的设计。

3-10 前完成类和函数的实现。

3-25 前完成模型训练。

3-31 前完成各功能的测试与补充。

4-5 前完成平台的部署配置和测试。

2.7 开发预算

项目	预算	备注
OPPO AI Unit	免费	感谢主办方提供的技术支持！
客户端手机	免费	感谢主办方提供的技术支持！
百度 AI 开放平台	免费	个人免费用户
主服务器	¥450	腾讯云服务器
计算节点	免费	个人电脑代替

3 可行性分析

3.1 技术可行性分析

目前深度学习技术在图像识别、AI 文字创作、图像修复和超分辨率等方面的应用较为成熟，也有很多成功的案例可以参考。并且性能开销均在可接受范围内，云端负责处理图片等数据量较大的内容，本地（手机端）处理文字生成等内容，可以降低总体处理耗时。深度学习技术在本项目中的应用是可以有效实现的。

云端的网络资源相当丰富，可以实时用关键词检索图片（或直接使用图片检索相似图片），检索相关表情包。并且云端检索的速度相对较快，耗时在用户可接受范围内。云端的数据检索也是可以实现的。

3.2 资源可行性分析

通用物体和场景识别：可以使用百度 AI 开放平台，调用图片识别 API。个人开发者每日 500 次免费识别请求，云端可以返回图片要素关键词和相关性得分。

自然语言主题和情感倾向分析：百度 AI 开放平台，个人免费用户每天不限请求次数。云端可以返回主题关键词和相关性及情感倾向打分。

根据关键词创作现代诗、歌词等有意境的配文：GitHub 有大量开源中文歌词（如 dengxiuqi/ChineseLyrics）、现代诗（如 qyxtim/modern-poetry）、古诗（如 THUNLP-AIPoet/Datasets）等开源数据集，有足够的可以用于模型训练。

图片的检索：可以使用百度图像搜索 API，可以根据关键词（或已有图片）检索全网相关图片。

表情包检索：GitHub 上有各分类的表情包数据集（如 zhaoolee/ChineseBQB 等）。并且可以将数据集上传至百度 AI 开放平台的自建库，调用自建库内的相似图片检索 API，可以获取相关的表情包。还可以在本地（手机端）进行添加文字等处理，可以得到能表达用户心情的表情包。

图像修复和超分辨率：OPPO AI Unit 有图像修复的 API，如果本地检测到图片（非表情包）分辨率过低，可以调用云端 API 进行图像处理。

3.3 市场可行性分析

当前朋友圈、QQ 空间、微博等用户动态分享空间已相当普及，许多人都想分享他们的生活与所想所感。但“词穷”现象很普遍，比如看到美丽的风景却想

不出具有美感和意境的词汇去形容。很多人甚至在朋友圈中表示想不出配什么文而直接发了图片。本项目可以帮助那些有表达欲的人群，他们个性化地生成配文以供参考。考虑到动态分享平台的用户基数很大，故本项目有一定的市场空间。

4 需求分析

4.1 数据需求

4.1.1 静态数据

用户数据：用户 id、用户密码、风格偏好

表情包：分类、含义、图片，用于查找相关表情

诗词歌词：分类、内容，用于训练配文模型

4.1.2 动态数据

用户收藏：用户 id、生成的动态编号

已生成的动态记录：用户 id、上传内容、返回内容

4.1.3 数据词典

用户列表：存储用户数据

表情列表：存储表情包数据

记录列表：存储用户上传的内容和返回的内容

诗词歌词表：分类、风格、内容

4.1.4 数据采集

用户数据：来源于用户注册和初始化时偏好选择。

表情包：来源于 GitHub 开源数据集（zhaoolee/ChineseBQB 等）。

诗词歌词：来源于 GitHub 开源数据集（dengxiuqi/ChineseLyrics、qyxtim/modern-poetry）等。

处理记录：来源于用户上传及处理完成记录。

4.2 功能需求

4.2.1 **功能模块

表 1 核心功能模块描述

功能模块	功能	功能描述	优先级
------	----	------	-----

文本预处理模块	关键词提取	提取用户给出的文本中的关键词	高
	情感分析	调用百度 AI 开放平台提供的情感倾向判断 API, 猜测可能的情感	中
图像预处理模块	主体元素提取	调用百度 AI 开放平台提供的通用物体和场景识别 API, 提取主体元素	高
	风格分析	通过分析主体元素, 联系主题和文字情感, 猜测图片要表达的意思。	中
素材检索与生成模块	图片检索	从网络中搜索符合主题与意境的图片	中
	文本生成	根据主题、内容、情感等元素, 并根据用户选择的风格, 生成类似古诗、近代诗、歌词等有意境的句子。	高
	表情包生成	从图库上获取合适的表情包, 并加以调整	低
	图片修饰	将模糊的图片或表情包进行修复, 并可以同时添加滤镜等艺术效果	低
账号信息模块	用户管理账户	存储相关偏好和收藏内容, 多端同步	低

表 2 文本预处理模块用例规约

用例名称	文本预处理模块
功能简述	根据文本分析情感积极性
用例编号	W001
执行者	用户
前置条件	① 用户输入文本 ② 成功发送到云端 ③ 调用云端百度 AI 开放平台 API
后置条件	云端返回此次文本分析的关键词和情感倾向
涉众利益	用户: 希望提供有效文本信息 用户: 希望额外获取符合的信息
基本路径	客户端-云端-百度 AI 开放平台-云端-客户端
扩展路径	云端出现错误-用户收到报错
字段列表	
设计规则	

未解决的问题	
备注	

表 3 图像预处理用例规约

用例名称	图像预处理模块
功能简述	提取图像主体元素，判断图片风格
用例编号	W002
执行者	用户
前置条件	① 用户上传图像 ② 成功发送到云端 ③ 云端调用百度 AI 开放平台 API
后置条件	云端保存此次图像分析的主体元素和风格倾向
涉众利益	用户：希望提供有效图片信息 用户：希望额外获取符合的信息
基本路径	客户端-云端-百度 AI 开放平台-云端-客户端
扩展路径	云端出现错误-用户收到报错
字段列表	
设计规则	
未解决的问题	
备注	

表 4 素材检索与生成模块用例规约

用例名称	素材检索和生成
功能简述	从资料库搜集或直接生成符合要求的图文资料，并加以改进。
用例编号	W003
执行者	云端
前置条件	① 已知用户希望的风格 ② 已知用户表达的情感和希望的朋友圈风格 ③ 已经获取到 W001、W002 的结果
后置条件	客户端收到并保存整个服务的结果
涉众利益	用户希望自动化完善朋友圈动态
基本路径	客户端-云端-图片检索 API/OPPO AI Unit-云端-客户端，或仅在客户端

	处理（文字）
扩展路径	
字段列表	
设计规则	
未解决的问题	根据信息生成图文的模型待训练
备注	核心

表 5 账户信息模块用例规约

用例名称	账户信息管理
功能简述	提供账号相关服务。保证个性化体验。
用例编号	W004
执行者	客户端
前置条件	① 用户注册或登录
后置条件	① 个性化的界面 ② 用户的风格偏好存储在云端 ③ 多平台同步账户
涉众利益	用户：个性化需求
基本路径	客户端-云端-客户端
扩展路径	
字段列表	
设计规则	
未解决的问题	
备注	可选模块，不登录也能使用 app

4.3 性能需求

4.3.1 时间特性

用户发起请求开始,到最终呈现结果,平均总响应时间小于 5 秒*图片数量。

其中, 占用时间较大的主要有: 数据传输(本地图文上传云端以及从云端下载生成结果); 神经网络迭代计算(本地和云端部署的 AI 框架的运算过程)。

4.3.2 适应性

云端部署的训练好的 AI 模型保持不变的情况下, 同时支持手机 app 端、微

信小程序端、PC 端、网页端的访问（暂仅支持安卓 app）。由于朋友圈等动态的制作需求集中在手机端，故手机端（安卓平台）是最重要的客户端，优先提供服务。所有端口都支持使用相应微信等社交软件客户端提供的 API 快速发送动态。

4.4 界面需求

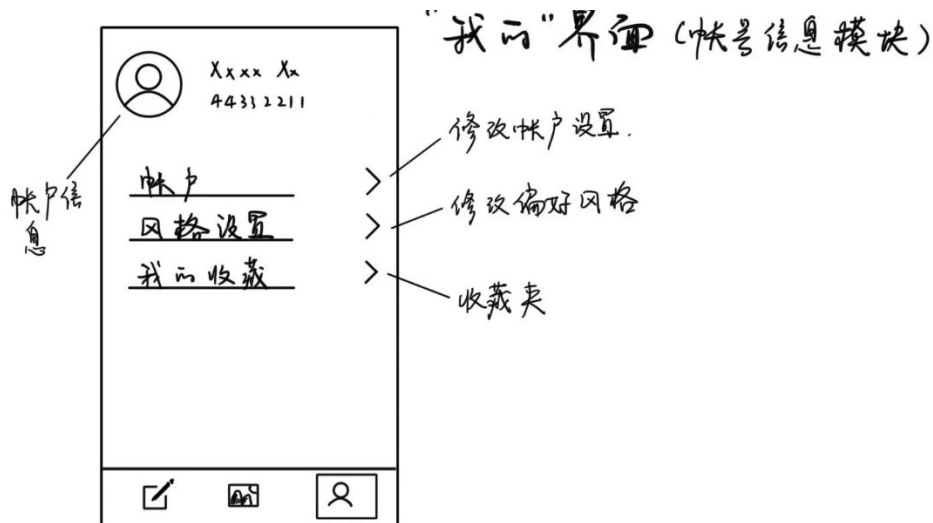


图1 “我的”界面 草图

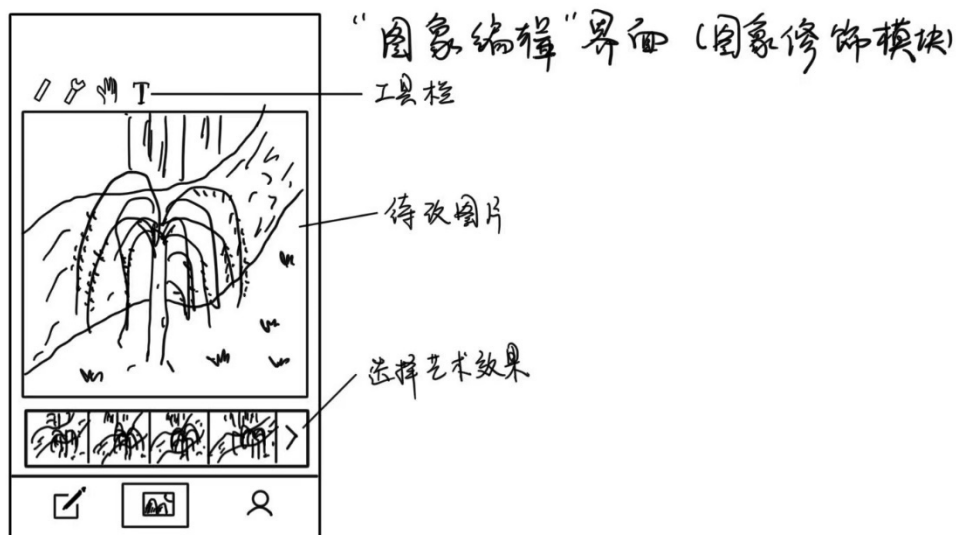


图2 图像编辑界面 草图

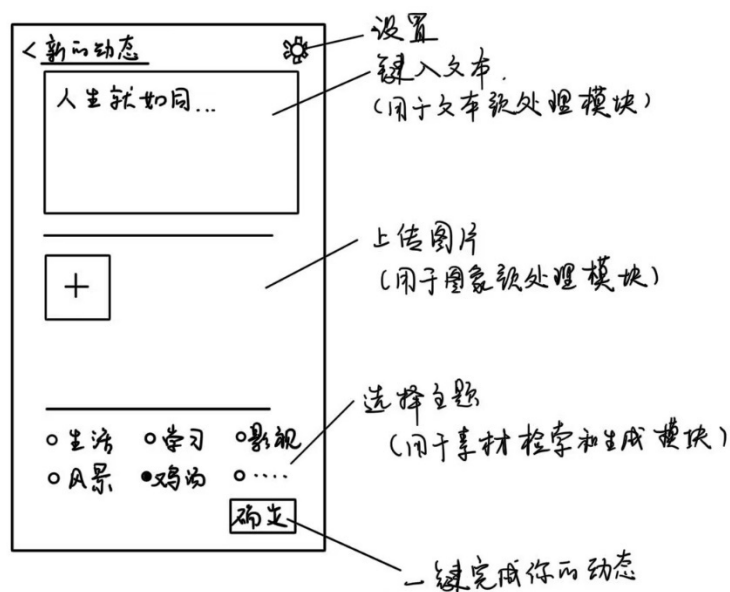


图3 “新的动态”创建界面 草图

4.5 接口需求

4.5.1 硬件接口

无

4.5.2 软件接口

百度 AI 开放平台中的通用物体与场景识别 API、语义和情感分析 API、OPPO AI Unit 照片修复 API（暂不需要）、百度图片搜索 API。

4.6 其他需求

可用性：95%正常使用时无故障发生。

安全性：保护用户隐私，用户数据隔离处理。

可移植性：手机、PC、网页等多平台通用（对应微信等社交软件客户端）

5 概要设计

5.1 处理流程

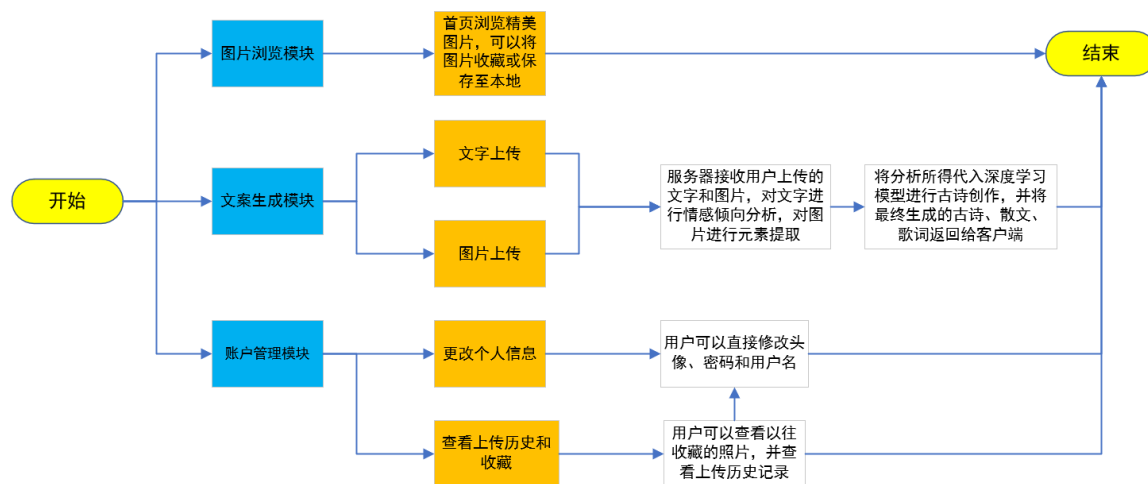


图 5.1.1 逻辑流程图

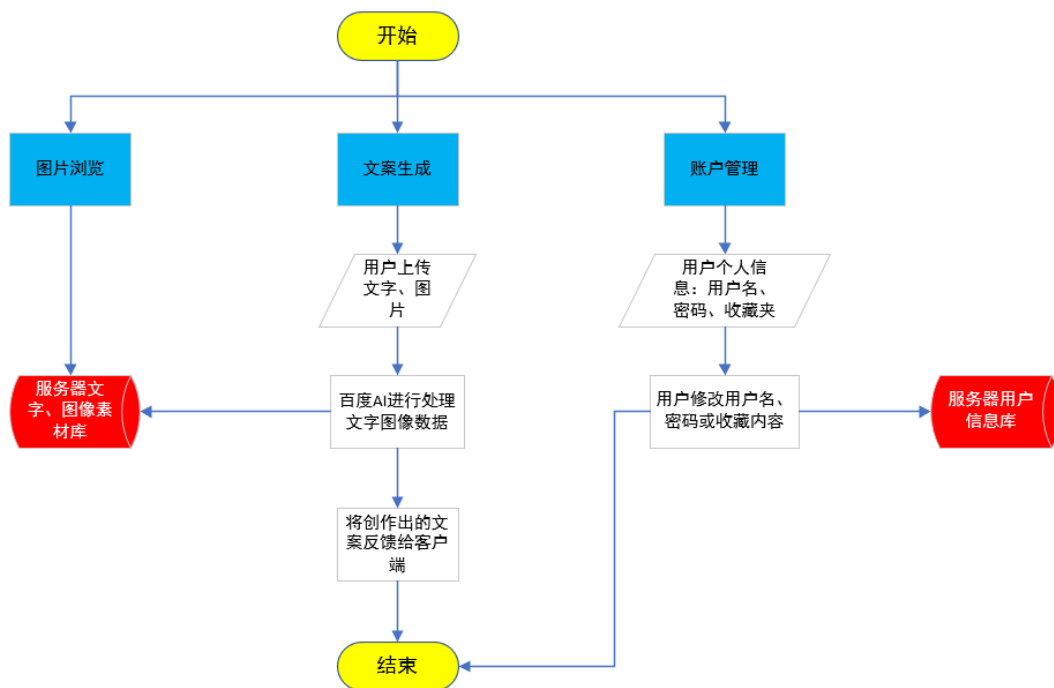


图 5.1.2 数据流程图

5.2 总体结构设计

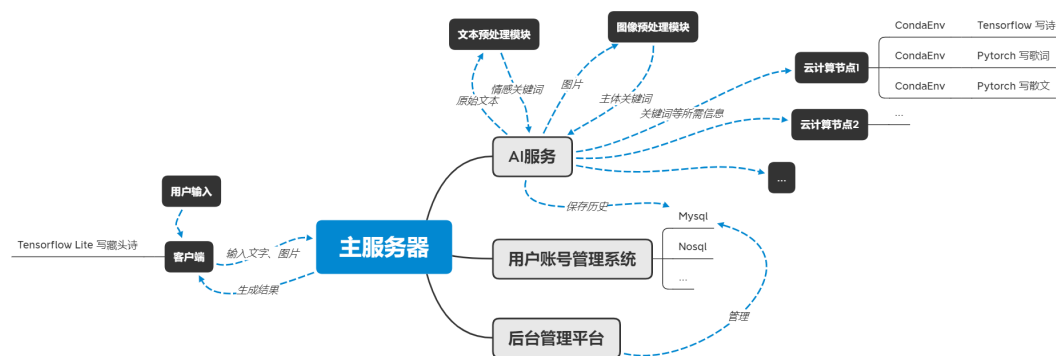


图 5.2 总体架构图

整体架构设计如上图所示，主要分为移动客户端和云端。其中客户端接受用户输入，在本地运行 TensorflowLite 模型生成藏头诗，分担云端运算压力，并把输入的文字和图片上传主服务器。主服务的 AI 服务模块将调用百度 AI 平台预处理文本和图片，提取情感倾向和图片主体关键词，并将预处理结果分配给云计算节点进行 AI 模型推断。云计算节点的文案生成部分主要涉及深度学习算法技术，利用预处理得到的信息进行运算并生成文案。并且云计算节点是可拓展的，可以适应高并发的用户请求，也可以保障可靠性。

同时，主服务器上还有用户账号管理系统和数据库，可以支持用户登录和信息同步。服务器上的后台管理平台，可以对上传侵权、违禁内容的用户进行封停和解封处理。

5.3 功能设计

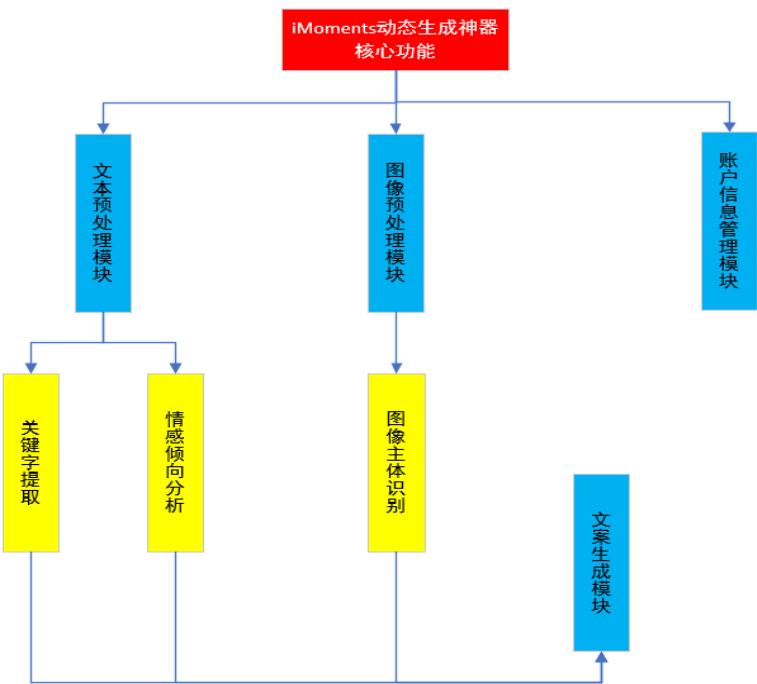


图 5.3 功能模块结构图

5.4 用户界面设计

1. 登录注册界面

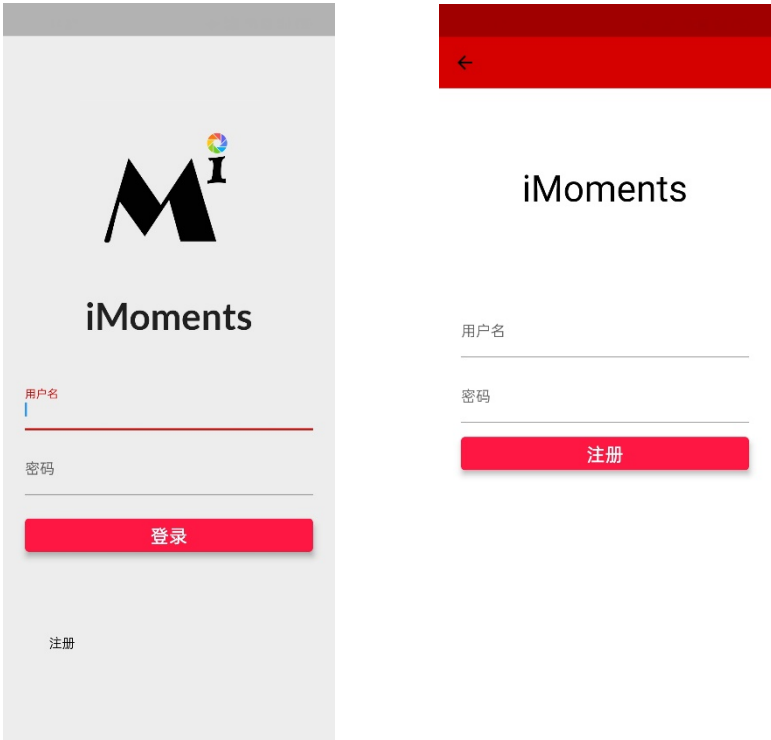


图 5.4.1 登录注册界面示意图

- 提供用户输入用户名、密码信息进行登录操作的图形界面接口。
- 提供用户输入用户名、密码等信息进行注册的图形界面接口。

2. 图片浏览模块

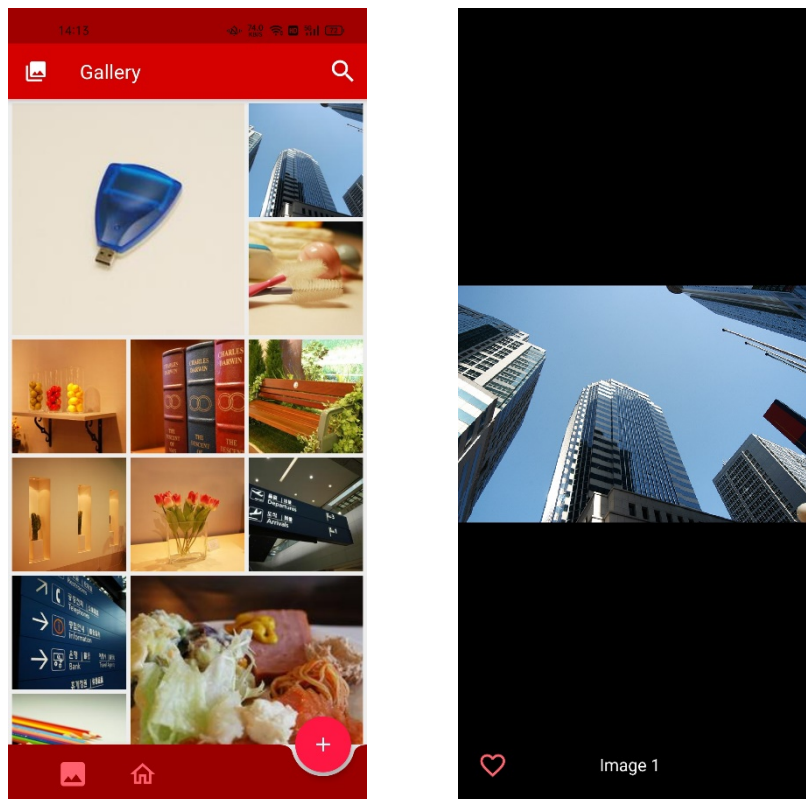


图 5.4.2 图片浏览示意图

- 提供用户浏览图片模式的图形界面接口。
- 提供用户通过点击图片来查看大图、收藏图片、保存图片至本地的图形界面接口。

3. 古诗生成模块



图 5.4.3 古诗创作生成示意图

- 提供用户上传图片的图形界面接口。
- 提供用户上传文字的图形界面接口。
- 提供用户得到生成的古诗句的图形界面接口。

4. 用户管理模块

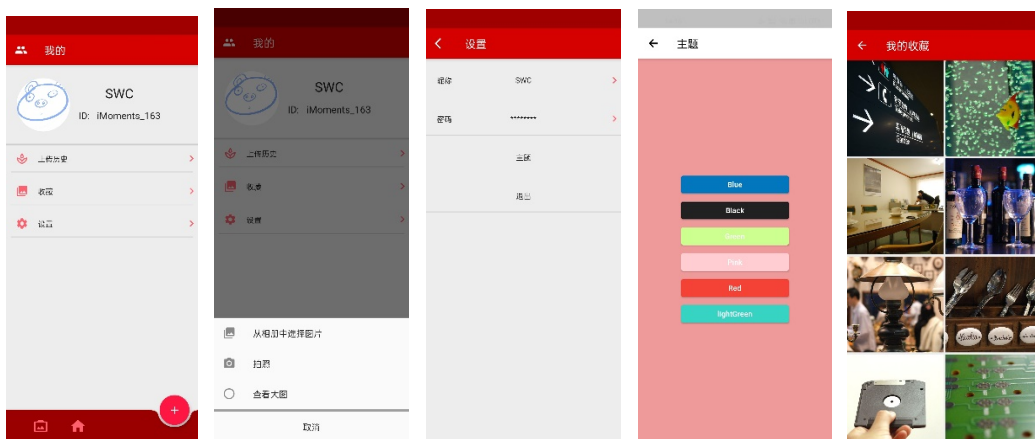


图 5.4.4 用户管理与收藏夹示意图

- 提供用户编辑头像的图形界面接口。
- 提供用户修改用户名、修改密码、修改主题的图形界面接口。
- 提供用户浏览收藏夹的图形界面接口。
- 提供用户退出登录的图形界面接口。

5.5 数据结构设计

前端：页面切换用栈来实现。

后端：逻辑层：

每个接口对应一个 handler，负责主要业务逻辑，调用模型层接口。

模型层：

把用户信息抽象成 User 类，调用数据层接口。

数据层：

提供操作数据库的接口，目前仅 Mysql，后续支持添加其它数据库。

大部分数据存在数据库中，数据库设计见第 6 节。

5.6 接口设计

5.6.1 外部接口

提供给外部（包括用户）使用的接口：

1. 注册和登录：
用户界面：图片见上文 5.4.1 节。
2. 修改账号信息：

用户界面：图片见上文 5.4.4 节。

3. 退出登录：

用户界面：图片见上文 5.4.4 节。

4. 查看精美画廊：

用户界面：图片见上文 5.4.2 节。

5. 收藏、保存图片：

用户界面：图片见上文 5.4.2 节。

6. 生成朋友圈：

用户界面：图片见上文 5.4.3 节。

使用第三方提供的接口：

1. 使用百度 AI 平台通用物体识别：

编写 Python 脚本，调用百度 SDK 实现。

2. 使用百度 AI 平台文字情感分析：

编写 Python 脚本，调用百度 SDK 实现。

5.6.2 内部接口

（下二图为服务端接口说明）

请求

类型：[method]
地址：http://1.15.86.128:8080/[service][query]
内容：[content]

服务	[method]	[service]	[content]	[query]
注册	POST	register	{" <u>username</u> ":"","password":""}	-
登录	POST	login	{" <u>username</u> ":"","password":""}	-
获取信息	GET	info	-	?token=_
更改信息	POST	info	{" <u>username</u> ":"","password":""}	?token=_
获取头像	GET	portrait	-	?token=_
上传头像	POST	portrait	图片的二进制数据	?token=_
图像分析	POST	graphic	图片的二进制数据	?token=_&type=_
新动态	POST	new	-	?token=_
文本分析	POST	words	文本内容	?token=_&type=_
管理员万能接口	POST	kill	-	?request=_

回应

状态码: [status]

内容: [content]

[status]	服务	[content]
200	注册	{"id": "_", "status": "register successfully"}
	登录	{"status": "login successfully", "token": ""}
	获取信息	{"id": "_", "status": "_"}
	更改信息	{"status": "update successfully"}
	上传头像	{"status": "upload successfully"}
	获取头像	图片的二进制数据
	图像分析	图片分析结果
	文本分析	文本分析结果
	新动态	{"serial": "_"}
	其他	-
400	ALL	{"error": ""}
401	ALL	{"status": ""}

5.7 错误/异常处理设计

5.7.1 错误/异常输出信息

错误/异常信息对我们程序的调试、完善有着重要的指导意义，但是我们不希望用户看到的错误/异常信息，所以我们认为，异常应该在控制/返回消息转发之前尽量处理完毕，同时发生异常的模块记录 log 日志，以便于我们开发者做进一步的调试和改进。此外我们希望部分异常不至于成为导致服务崩溃的“致命异常”，所以我们将各个层次建立尽可能完善的机制，将错误、异常信息有效的处理。

如果默认模块自有的错误/异常信息处理方式很可能会导致不必要的程序退出卡顿，所以我们专门建立了异常信息收集处理机制，我们将在数据层对其进行详细的介绍。异常信息收集器伪代码描述：

```
bool handleException(Throwable ex)
```

```
{
```

```
    保存错误/异常报告 log 文件;
```

```
    判断错误/异常类型;
```

```
        根据判断结果通知队列释放相关资源，并返回指示信息;
```

```
if (处理成功)
    return 1;
else
    强制释放该资源, 返回错误报告信息;
    return 0;
}
```

对于异常和错误信息的打印设置我们将进行如下展开:

1. 界面交互层

如果由于用户输入错误等将导致界面崩溃的问题, 我们通过 `try-catch` 机制进行捕获和转移, 并给出用户提示信息后, 根据需求重新加载界面, 并添加界面变化过渡效果。在异常处理伪代码中, 我们保存了错误的 `log` 信息, 并在系统中备份。同时我们通过相关线程释放资源, 避免应用资源的浪费; 同时返回给接口调用者错误类型信息使其变更操作, 以此保证用户界面良好的交互性和设计原则。

2. 逻辑层

该部分我们将详细介绍逻辑层核心模块和其他主要模块的异常信息处理机制。

对于核心模块部分, 大部分异常来源于用户的非法上传信息, 首先我们在前端设置了第一层限制, 保障我们后台得到的数据干净有效。但是如果仍然存在数据来源问题引发的异常, 为了避免此类情况的而发生, 我们在与文字预处理、图像预处理模块都留有异常捕获机制, 如果发生时间超时, 图像、文本不符合要求等问题, 我们都会事先判断是否会出现问题, 并且给用户相应的提示, 保证用户良好的使用体验。对于核心模块, 我们的分块, 分区域性明显, 每个部分各司其职, 当任何部分出现异常, 会产生相应的消息并且会使用临时变量作为代替, 以防产生连锁反应, 达到了高效、并行的效果。如: 当用户上传的图像无法解析, 我们会使用事先的默认空白图像作为默认值返回代替, 并且提示用户图像解析出现问题。

3. 数据层

数据层部分的异常信息主要是数据类型、数据范围以及约束条件等。该部分的异常信息在一定程度上需要返回给接口调用者, 以便于指导用户的数据修改和完善, 我们在捕获异常信息之后会对异常类型进行解析处理, 转换为我们专用符号, 这些专用符号将在上层翻译为用户易懂的指导性更正意见。

5.7.2 错误/异常处理对策

1. 界面交互层

对于运行中出现的异常和错误, 我们设置应用线程调度器的线程调度问题如

果出现异常且在不可控范围内，则使用缓存数据，同时释放资源，创建新的线程去重新处理失败的任务部分。

2. 逻辑层和数据层

对于运行中出现的异常和错误，如果在不可控范围内，则丢弃该任务，同时释放资源，并将错误 log 信息在系统中备份，同时返回接口调用者错误类型信息使其变更操作。

5.8 系统配置策略

客户端：

Android SDK 最低版本 16，目标版本 30

表 5.8.1 客户端机型配置表

手机型号	OPPO Reno5 Pro+ 5G
Android 版本	11
系统版本	ColorOS V11.1
处理器	Snapdragon 865 5G Mobile Platform
运行内存	8.00GB

云端服务器：

Master:

表 5.8.2 主服务器配置表

CPU 型号	Intel® Xeon® Gold 6148 CPU
CPU 架构	x86_64
CPU 核心数	2
CPU 线程数	2
CPU 主频	2.4GHz
运行内存	4.0GB
网络带宽	3Mbps

Node:

表 5.8.3 计算节点配置表

CPU 型号	Intel® Core™ i7-9750H CPU
CPU 架构	X86_64
CPU 核心数	6

CPU 线程数	12
CPU 主频	2.6GHz
运行内存	15.8GB
网络带宽	1Gbps
GPU 型号	NVIDIA GeForce RTX2070 Max-Q
GPU 显存容量	8.0GB

5.9 系统部署方案

客户端：打包好的 apk 直接在手机上安装

主服务器：部署在腾讯云，具有公网 ip，接收客户端请求，并向计算节点分配计算任务。

计算节点：有三个 Conda 虚拟环境，分别负责各模型的运行。

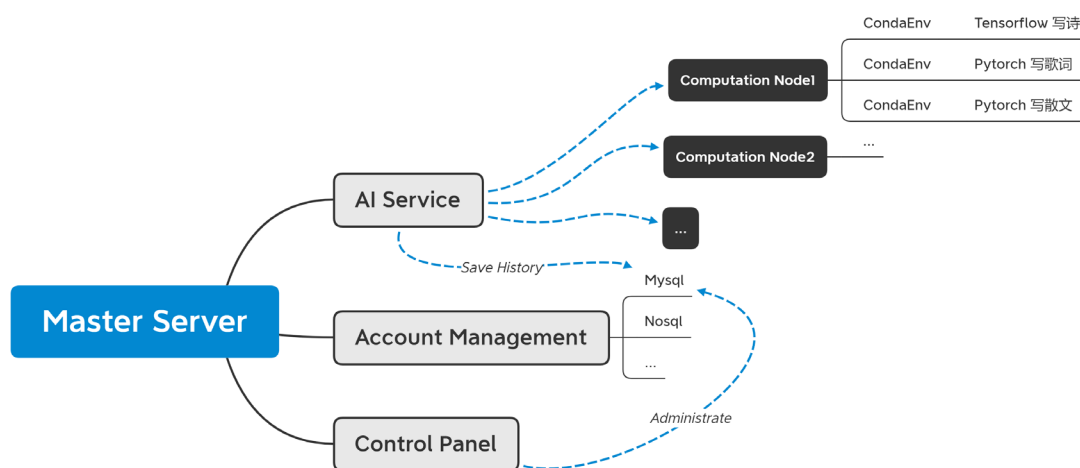


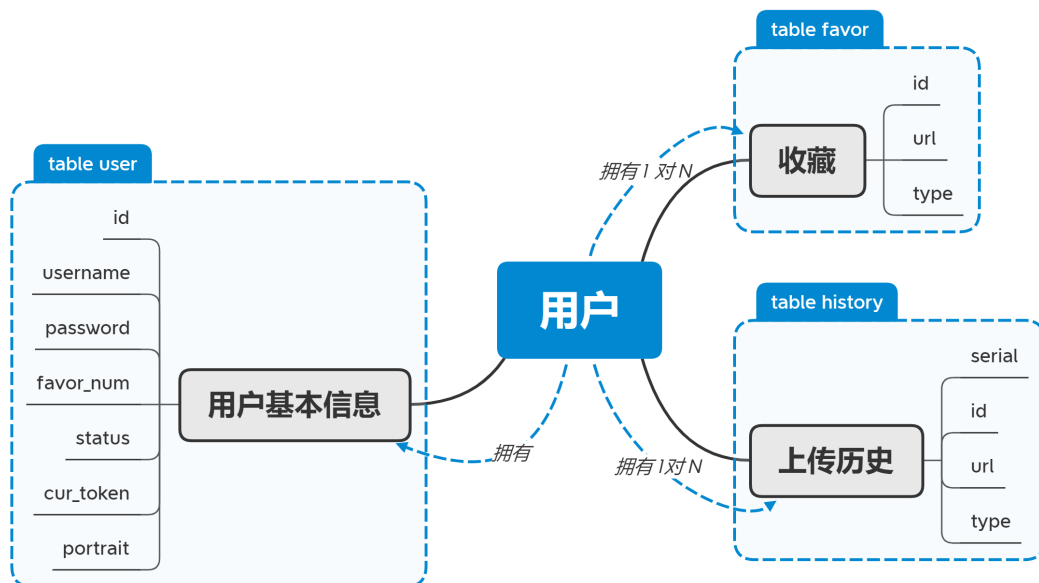
图 5.8.1 系统部署方案图

5.10 其他相关技术与方案

无

6 数据库设计

(下图为 ER 图 使用 XMind 制作)



(下图为后台 mysql 数据信息)

```
mysql> show tables;
+-----+
| Tables_in_test |
+-----+
| favor          |
| moments       |
| user          |
+-----+
```

```
mysql> select * from user;
```

id	username	password	status	token	portrait	favorNum
1	LeC	192277	n	HD4IRHGVP00ZSYVCTNLMSPQOHRQXNBFROZBEJINXEYDNWHLZLYFNAGCDKIOQJM	WGU4PEAENSQPNBYOAWRPMILXVCMFPMWI.jpg	1
2	HuID	200176	f	RK1GBUGSCBTPDOHVFGTICXEOGHRCEXPPPMGGEATYKMLUCHPKOREMUCRIZVHH	LAXFMGNVWJBTBAVDQZWAJTSEMYILRAZ.jpg	0
3	Awel	123456	n	FTZSOKDMEACPLNNVFNUATXDNWZNTFDENHVVJQDKSRKXSNRQFUHDXWHNMEBGES	LVRVERCKKPAQKYLILANZQOOFRAJBHAJ.jpg	2
4	BigJ	665566	n	NULL	default.png	0
6	tyt	6282673	n	NULL	default.png	0
7	tit	62726393	n	NULL	default.png	0
8	lcl	62726393	n	NULL	default.png	0

(上图账号密码均为测试用户)

Id: 自增整型 每个用户唯一的 id 号

Username: 不可重复字符串 用户名

Password, Token, Portrait: 字符串 密码, 服务密钥, 头像地址

Status: 字符 帐号状态

FavorNum: 整型 收藏夹大小

```
mysql> select * from moments;
```

serial	id	datetime	type	path
1	2	2021-04-04 23:10:16	1	https://www.imoments.com.cn/uploads/CKFCYFNDCMWZCPCWQVMAKPHDSQOPGGVU.txt
1	2	2021-04-04 23:10:24	0	https://www.imoments.com.cn/uploads/YQIZBGEJGIJAWBEKEDQDQSPYGARKHVIS.jpg
1	1	2021-04-04 23:45:35	0	https://www.imoments.com.cn/uploads/TMMJKNUPHIGQTMJPDHDCAOHABPARZJH.jpg
1	1	2021-04-04 23:45:39	0	https://www.imoments.com.cn/uploads/NZCKJOPAHTBUVVJNLEQSLZERMRKELLOK.jpg
2	148	2021-04-05 00:03:12	1	https://www.imoments.com.cn/uploads/DEBBDGZYBADRWZIKPAUQJVBVCCJSMWE.txt
2	148	2021-04-05 00:03:19	0	https://www.imoments.com.cn/uploads/NHMOKQKKJDXZCULEZBRIBHMXOUREMPYI.jpg
1	92	2021-04-05 00:05:27	1	https://www.imoments.com.cn/uploads/ZGUPDCLBOJZDVVVATVQONJPRUMKNJAW.txt
1	92	2021-04-05 00:05:34	0	https://www.imoments.com.cn/uploads/ABSBHVXCCAQABAGHVTRLUXXTGTVDTSXM.jpg
2	1	2021-04-05 00:08:50	0	https://www.imoments.com.cn/uploads/DWYTDXXTFNRFOMQSJBGXGXZJSTIMBSDI.jpg
2	14	2021-04-05 00:08:52	0	https://www.imoments.com.cn/uploads/UPBEXHETGKANJOPBYGEMAAALNGVTAIKCY.jpg

(上图 path 字段中链接可以直接访问并打开,用于管理员审核上传内容)

Serial: 整型 每条动态的编号,一个动态可能对应多条记录

Id: 整型 用户 id

Datetime: 日期时间类型 发送时间

Type: 布尔型 类型是否为图片

Path: 字符串 内容地址

```
mysql> select * from favor;
```

id	imgurl
1	https://www.imoments.com.cn/resource/img/1/542.jpg
3	https://www.imoments.com.cn/resource/img/1/1307.jpg
3	https://www.imoments.com.cn/resource/img/1/1347.jpg

(上图 imgurl 字段中链接可以直接访问并打开,用于管理员审核上传内容)

Id: 整型 用户 id

Imgurl: 字符串 内容地址

7 手机端侧部署设计

7.1 手机环境需求

Android:

minSdkVersion: 16

targetSdkVersion: 30

要求使用权限: 网络、读写内存

7.2 调用 AI Unit 能力接口设计（可选）

7.3 AI Boost 端侧模型加速设计（可选）

备注:

由于项目使用的是 flutter 框架, OPPO AI Unit 也没有 flutter 支持, 所以很遗憾之前设想的照片降噪修复等调用 AI Unit 和 AI Boost 的功能可能无法实现。

8 详细设计

8.1 文本预处理功能模块

8.1.1 功能描述

提取用户输入文本中的关键字信息，并调用百度 AI 开放平台提供的情感倾向判断 API 进行情感分析，给出文字的“积极”程度。

8.1.2 性能描述

采用的百度 AI 提供的情感倾向分析 API，自动判断该文本的情感极性类别并给出相应的置信度，情感极性分为积极、消极和中性。

时间：经测试，百度平台会在 2 秒内返回结果。

8.1.3 输入

段落、长句或关键词

8.1.4 输出

情感倾向

8.1.5 程序逻辑

文本输入 → 关键字句提取 → 调用 API 进行分析 → 情感倾向

8.1.6 限制条件

输入为中文。输入不得超过 100 字。

8.2 图像预处理功能模块

8.2.1 功能描述

调用百度 AI 开放平台提供的通用物体和场景识别 API，提取图像主体元素，并通过分析主体元素，联系主体和文字情感进行风格分析。

8.2.2 性能描述

采用的百度 AI 提供的通用物体和场景识别 API，识别动物、植物、商品、建筑、风景、动漫、食材、公众人物等 10 万个常见物体及场景，并向用户返回大类及细分类的名称结果。

时间：百度平台响应速度取决于图片大小和网络质量。在 100M 带宽网络、1M 以内图片条件下，会在 5 秒内响应。

8.2.3 输入

用户图片

8.2.4 输出

提取出的主体元素关键词

8.2.5 程序逻辑

图像输入 → 图像压缩 → 图像提取分析 → 图像元素关键词

8.2.6 限制条件

输入图片每张大小不超过 5MB。

输入图片为用户图片不能涉及版权问题。

输入图片内容应符合社会主义核心价值观，不能上传违规违法图片。

8.3 素材检索与生成功能模块

8.3.1 功能描述

从云端上搜索符合主题与意境的相似图片，并根据文本情感与图像主体元素，结合深度学习算法写出有意境有美感的古诗。

8.3.2 性能描述

快速进行云端搜索，运用卷积神经网络模型进行深度学习创作出好的古诗、优美的散文、朗朗上口的歌词。

时间：诗歌会在 1 秒内给出，散文、歌词会在 1.5 秒内给出。

8.3.3 输入

文本预处理模块限制条件和图像预处理模块给出的关键词和情感

8.3.4 输出

AI 创作出的诗、文

8.3.5 程序逻辑

模型运行 → 关键词、情感词输入 → 类型选择 → 输出生成结果

8.3.6 限制条件

文本预处理模块限制条件和图像预处理模块正确给出关键词和情感。

8.4 账号信息功能模块

8.4.1 功能描述

存储用户 id、密码、相关偏好和收藏内容，实现多端同步。

8.4.2 性能描述

运用 flutter 编写的前端响应，将用户输入实时发送给后端服务器，并在前后端之间不断交流通信，实现客户端请求云端接收并返回，从而实现多端同步。

8.4.3 输入

请求类别，请求参数

8.4.4 输出

是否成功，其他返回值

8.4.5 程序逻辑

用户输入 → 前端发送请求 → 服务器接收请求并更新数据 → 服务器返回输出

8.4.6 限制条件

网络通畅，服务器空闲

附件：最新 release 版本下载 (.apk):



若无法登录，可能是服务器处于维护状态。