

系统开发说明书

1. 产品设计方案

1.1. 项目实施可行性报告

1.1.1. 行业市场分析

目前市场上的即时通讯产品主要有微信、QQ 等，就通讯服务应用程序而言，市场接近饱和。主流通讯服务应用程序不仅能提供稳定便捷的文字、视频通信服务，还可以通过多元化的附加服务满足用户日常的消遣、社交需求。然而，目前主流的通讯产品的通信都需要专门设置的服务器支撑，换句话说，通信的数据都需要传输到服务器，再由服务器发送到接收端。收发信息无论是安全性还是速度效率在一定程度上不可避免地受到服务端的限制。相反，不需要服务器支撑、让通信节点同时充当客户端与服务端的通讯服务应用产品在市面上数量较少。

1.1.2. 竞争对手分析

与服务多元化发展的市面上主流通讯服务应用程序不同，本产品主要聚焦于通信的隐私与可靠性，这将意味着其能在与其他广泛使用的即时通讯产品的竞争中能够“扬长避短”。

产品	微信	QQ	本产品
功能	文字、语音（包括强联网方式与弱联网方式）、视频通信、小文件传输等，附有一定的娱乐、社交服务	文字、语音、视频，大小文件传输，附有大量的会员增值服务、娱乐、社交项目等	弱联网语音通信、文字消息、文件传输等

优势	功能齐全、方便快捷 目前用户数量庞大，用户粘性极大	大小信息交换便捷、服务功能多元化、用户粘性较大	不需要中间服务器支持，在通讯的安全性、可靠性上不再“受制于人”
劣势	收发数据受服务器的诸多限制、如服务器崩溃、数据丢失等等 第三方插入广告等	对于通讯程序来说，功能略显臃肿 收发数据依赖服务器	功能多元化欠缺、通信速度有限等
受众人群	普通用户	普通用户	普通用户、高隐私性需求的用户

1.2. 产品定位及目标

产品定位为一款面向会使用智能手机的群体，集文字聊天、语音聊天、发送文件和图片等功能于一体的可操作性强、页面美观、性能完好的高质量即时通讯软件。

产品目标为经过推广后能稳健发展为一款能满足广大群众交流、学习及办公需求的即时通讯软件。

1.3. 产品内容总策划

1.3.1. 应用流程规划

- 需求分析：通过小组讨论，确定 app 的定位，进行功能模块划分并进行系统的逻辑设计，研究分析需求实现的可行性，确认每个需求的实现逻辑。
- UI 设计：根据产品的运行平台和实现模式进行人机交互界面设计，设计出的布局、控件、配色等能使 app 最终的 UI 效果图美观整洁。
- 后端设计：设计系统整体架构，建立 app 数据库，设计需要用到的数据结构和算法，以及接口封装和后台管理。
- 系统编码：进行前端页面编码，特效制作；总框架编码，各功能模块编码，以及各功能模块内部测试。
- 系统测试：编写测试计划，包括测试范围、测试环境、测试方法、测试用例、

测试场景等。按测试计划对系统进行测试，记录软件缺陷。

1.3.2. 设计与测试规范

制定规范的目的是确保设计的统一性与合理性。规范是为了项目利益最大化，高效化而在团队中制定成的约定。

● UI 设计规范

- a) 界面尺寸：在进行 UI 设计时，首先要根据安卓界面尺寸来确定画布大小，目前安卓端主流尺寸主要是@1.5x，@2x 以及@3x。同时建立好状态栏、导航栏、菜单栏等参考高度。
- b) 系统图标：严格遵守图标大小，图标设计，图标切图规范，以及切图命名规则。
- c) 字体：安卓系统中，中文使用的是谷歌思源，英文使用的是 Roboto。若选用其他字体，必需考虑所选字体是否利于用户阅读。对于字体的大小，在界面设计过程中，需要统一，比如所有正文统一大小，所有标题统一大小。

● 后端设计规范

- a) 模块化开发：不同功能模块之间尽量独立，职责分明。原则上不能在本模块中实现其他模块的功能，这有利于以后做系统拆分以及系统测试。
- b) 面向对象：充分利用继承多态封装的特性进行解耦；接口最简化，与外部对接的方法、类、参数尽可能简单，没有冗余；公开方法最少化，其他类能调用本类时能调用的方法尽可能少。
- c) 代码书写规范：统一编码、统一缩进、空格的使用等，提高代码整洁度。
- d) 命名规范：包名遵循域名反转规则；类命名使用驼峰命名法，用名词或名词词组命名，每个单词的首字母大写；方法命名使用驼峰命名规则，用动词命名，第一个单词的首字母小写，其他单词的首字母大写。变量的命名需要与其功能相对应。

● 软件测试规范

- a) 尽早地和不断地进行软件测试。不应把软件测试与 app 开发独立开来，而应当把它贯穿到开发的各个阶段中。
- b) 测试用例应由测试输入数据和与之对应的预期输出结果这两部分组成。
- c) 尽量避免自己检测自己的代码，让组员来测试自己编写的程序。
- d) 在设计测试用例时，应当包括合理的输入条件和不合理的输入条件。妥善保存测试计划，测试用例，出错统计和最终分析报告，为维护提供方便。

1.3.3. 开发日程表

完成日期	阶段	主要负责人
2021-10-21	需求分析	全组成员
2021-11-21	UI 设计	罗明炜

2021-11-28	区块链底层开发	黄承鹏、罗明炜
2021-11-28	即时通讯模块开发	陆泓相、莫怡灿
2021-12-12	整合所开发模块	全组成员
2021-12-21	系统测试	陆泓相、莫怡灿
2021-12-30	撰写结题报告	全组成员

1.4. 技术解决方案

1.4.1. 开发平台

开发平台主要为 Windows + Android Studio，使用语言为 Java 和 Kotlin。

1.4.2. 测试平台

在真机和模拟器上对产品进行测试。

1.4.3. 技术架构

- 技术架构自底向上分别为存储层、数据层、网络层、协议层、共识层、应用层。
- 存储层：包括本地文件系统和 LevelDB 数据库。
- 数据层：包括区块、区块链、Hash、Merkel 树、非对称加密。
- 网络层：P2P 对等网络。
- 协议层：HTTP、RPC 等。
- 共识层：POS 或 DPOS。
- 应用层：即时通讯模块。
- 展示层：UI 界面。

1.5. 推广方案

1.5.1. 产品展示

在学术环境中进行产品展示，在学术界充分探讨该产品的功能、个性以及优缺点。吸取多方意见对产品进行改进，细化用户需求，对产品进行针对

性优化等。

1.5.2. 指定范围内部测试

发展指定范围内小部分用户与测试人员一起使用，对使用过程进行跟踪与分析，就分析结果与反馈体验对产品进行进一步改善。

1.5.3. 小规模推广

将本产品向在需求分析基础上的潜在用户（以通信安全与可靠为第一考量的特殊需求的用户）推广。

1.5.4. 宣传推广

重点在在网络环境中将本产品进行全面推广（主要利用宣传视频在各大短视频应用中推广与设计图文让本产品进入网络论坛），线下进行海报张贴、用户发掘、市场调研等工作。

1.6. 运营规划书

1.6.1. 广告服务

产品中设计有开屏广告和横幅广告，其中开屏广告为打开产品时展示的广告；横幅广告为聊天消息窗口顶端展示的可关闭广告。

产品可交付使用后，计划先按照产品使用人数、发展前景等条件接收广告；待产品发展到一定规模，可使用按展示效果收费方式收取广告费用。广告服务收取的费用将用于产品的维护与升级。

1.6.2. 打赏

为解决产品后续维护和升级的资金问题，产品中提供“打赏”窗口供欣赏此产品的用户打赏开发团队任意金额的途径。

1.6.3. 风险评估

- 技术风险：产品涉及区块链、P2P 等技术，团队成员的技术水平可能不足，可能不能按预期目标完成产品的设计与开发。
- 政策风险：用户的聊天内容不受监管，可能涉及违法违规信息，可能对社会造成影响，可能不符合法律法规精神。
- 运营风险：用户数量可能不足以支撑 P2P 网络，可能形成数据孤岛。

2. 产品实现方案

2.1. 系统的主要功能

2.1.1. 登录注册

- 1) 注册：用户在第一次使用本 APP 时可以注册新 id，同时设定登录密码。
- 2) 登录：用户在每次使用本 APP 前需输入注册时设定的 id 和登录密码，验证通过才可使用本 APP。

2.1.2. 聊天管理

- 1) 置顶聊天：用户可以把特定的好友聊天置顶在自己的聊天列表中。
- 2) 消息免打扰：用户可以设置对特定的用户进行消息免打扰。
- 3) 发送信息：用户在和好友聊天时可发送文本、图片、语音。

2.1.3. 通讯录管理

- 1) 修改备注：用户可以为自己的好友设置备注。
- 2) 删除好友：用户可以删除自己的好友，同时会删除与其的聊天信息。
- 3) 好友请求：用户可以查看其他用户向自己发送的好友请求，选择同意或拒绝。
- 4) 添加新好友：用户可以搜索其他用户，并向对方发送好友请求。

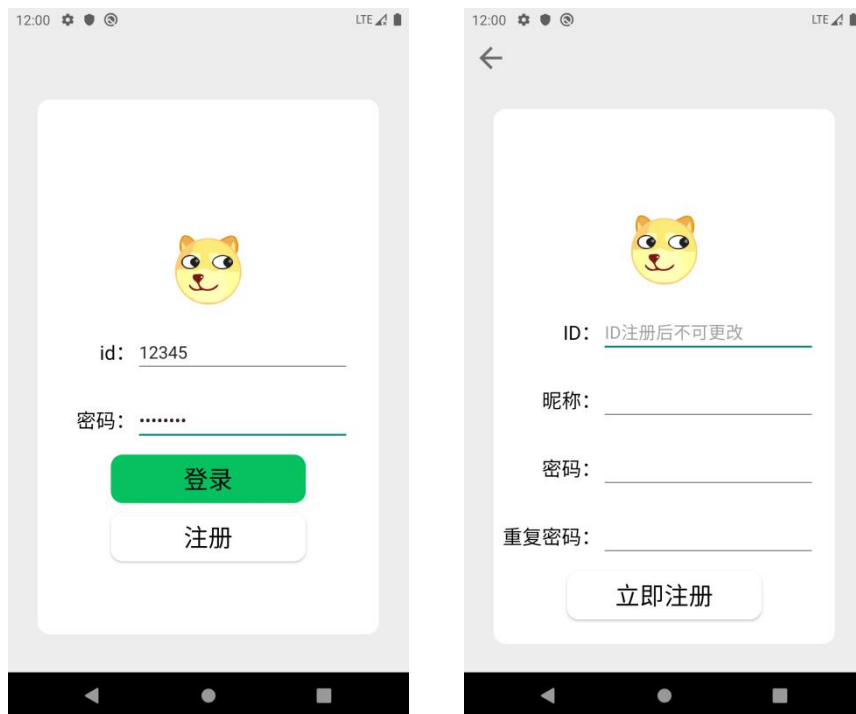
2.1.4. 设置

- 1) 设置昵称：用户可以在设置中修改自己的昵称。
- 2) 修改头像：用户可以从相册中选择图片或通过手机摄像头拍照进行修改头像。
- 3) 修改密码：用户可以输入原密码和新密码进行修改密码。

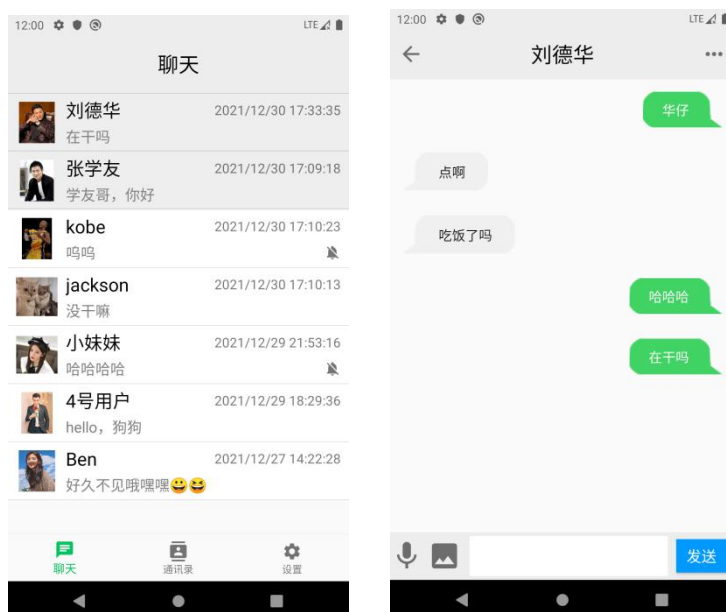
4) 退出登录：可以退出当前用户，登录其他用户或退出 app。

2.2. UI 界面设计

2.2.1. 登录注册界面

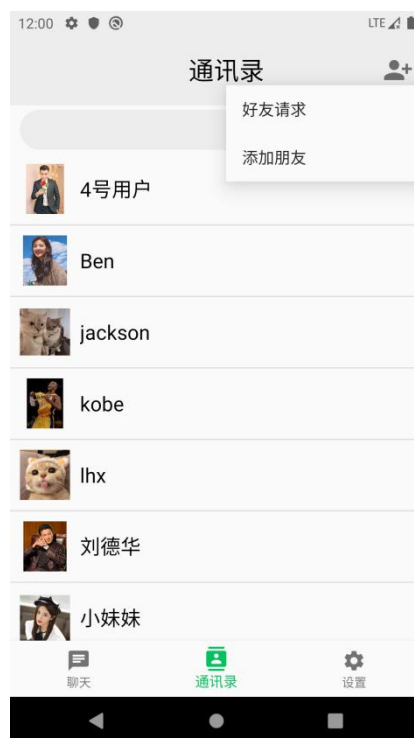
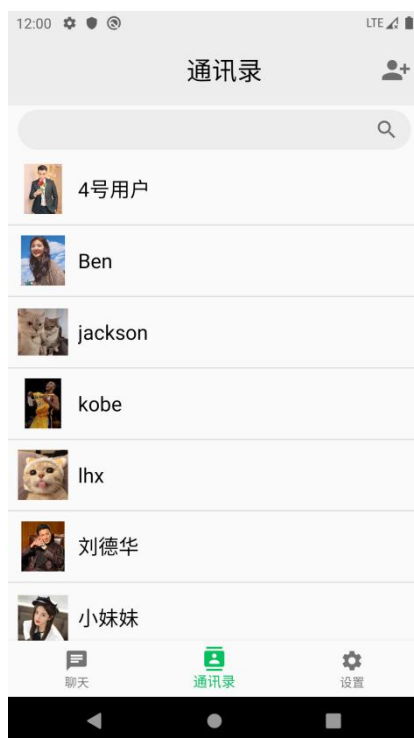


2.2.2. 聊天界面





2.2.3. 通讯录界面





2.2.4. 设置界面





2.3. 关键技术和技术难点

2.3.1. 关键技术

本项目采用了最基本的基于 Kotlin 语言的 Android 开发技术，开发方案是把项目功能分为注册登录模块、聊天模块、通讯录模块、设置模块的四大模块。对于每个界面都实现一个布局文件，然后为其设置对应的 Activity，在 Activity 中实现相应的功能。

数据存储采用了 SQLite 数据库，设计的数据库如下几个表：

user（用户表）：

属性	类型	描述
id (primary key)	text	用户的 id
password	text	用户的密码
name	text	用户的昵称
profile_photo	BLOB	用户的头像的二进制形式

friendship（好友关系表）：

属性	类型	描述
ownerID (primary key)	text	主用户的 id
friendID (primary key)	text	好友的 id
note	text	好友备注

friend_request（好友申请表）：

属性	类型	描述
applicantID (primary key)	text	请求方的 id
respondentID (primary key)	text	回应方的 id

chat（聊天列表）：

属性	类型	描述
ownerID (primary key)	text	主用户的 id
friendID (primary key)	text	好友的 id
lastMessage	text	最后一条消息的内容
lastTime	text	最后一次发消息的时间
stickOnTop	integer	0 为不置顶, 1 为置顶
msgIgnore	integer	0 为不消息免打扰, 1 为消息免打扰

msg_content (聊天内容表):

属性	类型	描述
ownerID (primary key)	text	主用户的 id
friendID (primary key)	text	好友的 id
context	text	消息内容
msgType	integer	消息类型, 0 为接收, 1 为发送
sendTime	text	发送时间

2.3.2. 技术难点

在实现主界面的时候, 采用了一个 MainActivity 与三个 Fragment 绑定, 分别是聊天界面 ChatFragment、通讯录界面 ContactFragment、设置界面 SettingFragment。由于书本上对在 Fragment 中实现点击事件和跳转等的讲解不是很全面, 再而就是网上关于 Kotlin 的资源较少, 所以在初步实现三大模块的绑定和功能的时候遇到了一定的困难。

2.4. 用户体验记录和分析

2.4.1. 用户体验记录

体验用户	用户反馈
用户 1	免打扰好评
用户 2	密码竟然能设置 1 位
用户 3	界面都是白色的，有些单调
用户 4	有些编辑框编辑文字时文字太贴近编辑框边缘
用户 5	感觉是简约版的微信
用户 6	支持表情发送就更好了
用户 7	不支持发送语音、图片
用户 8	可以设置搜索好友功能用于搜索现有好友

2.4.2. 用户体验分析

产品功能基本已实现，用户反馈中更多的是对产品的建议。总结分析本产品在一些细节上仍需要继续改进，比如密码应该设置位数限制、UI 设计需要继续美化等，而且应该做出自己产品的特色。

2.5. 已完成的改进和存在的问题

2.5.1. 已完成的改进

改进 1：增加了开屏动画功能，使产品更自然美观，且可以用于接受开屏广告。

改进 2：注册时的限制条件，ID 与昵称不能含有空格，密码大于等于 8 位。

改进 3：在聊天、好友页面使用 SwipeRefreshLayout 实现下滑刷新功能。

改进 4：在编辑框设置 padding 属性填充边缘防止文字贴近边框。

改进 5：在通讯录页面新增通过好友 ID 或用户名全部或部分内容搜索现有好友的功能。

改进 6：使用 AlertDialog 的 GetWindow 功能获取 AlertDialog 页面，把默认按钮样式与方形边框更改为圆角边框，美化页面显示。

改进 7：把产品的默认方形按钮更改为圆角按钮，使页面更美观。

改进 8：其他使页面更美观的改进。

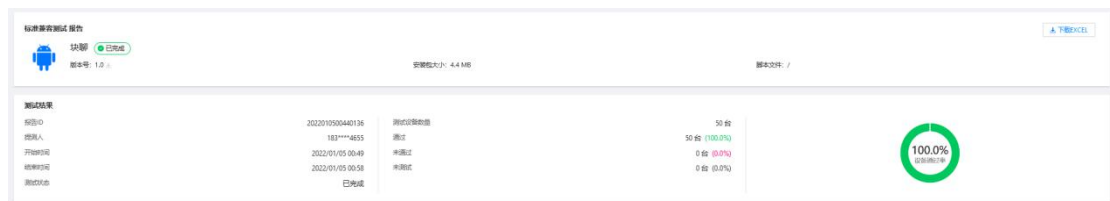
2.5.2. 存在的问题

问题 1：更改头像时没有剪切图片大小的功能。

问题 2：好友聊天时不能发送图片。

3. 测试大纲和测试报告

兼容性测试，对 Android 机型使用量 TOP50 的机型进行兼容性测试，测试结果如下



4. 产品安装和使用说明

4.1. 产品安装说明

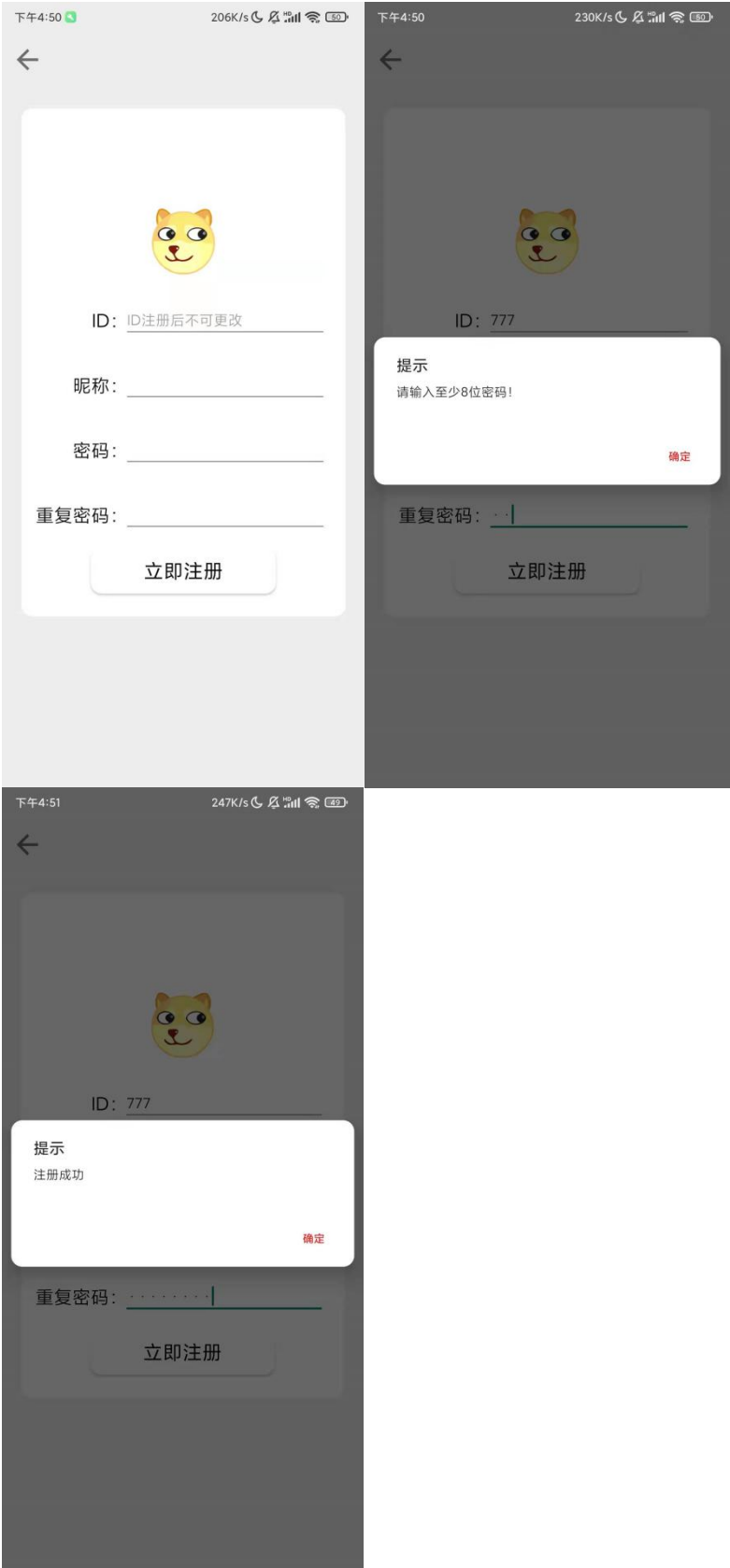
本产品为安卓应用软件,用户把产品安装包下载保存到安卓平台的智能手机上后便可进行安装,操作简单无注意事项。

4.2. 产品使用说明

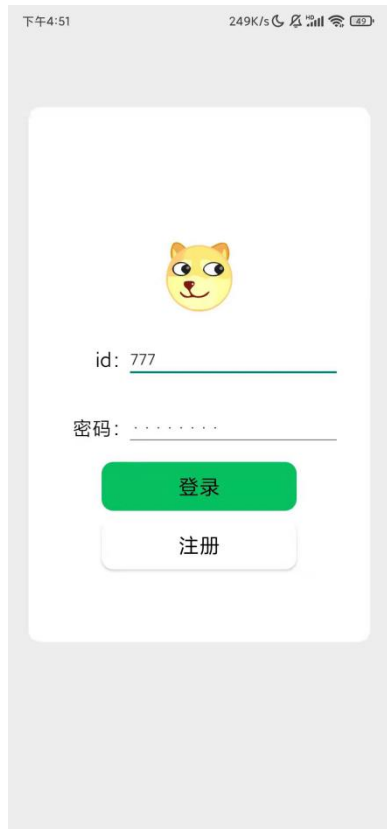
4.2.1. 注册与登录

安装本产品后用户需要使用 ID、昵称、密码和重复密码进行注册后方可使用，其中为

保证账户安全设有限制条件，ID 与昵称不能含有空格且不能超过 15 位，密码含有至少 8 位。



注册完成后用户使用 ID 与密码便可登录系统。

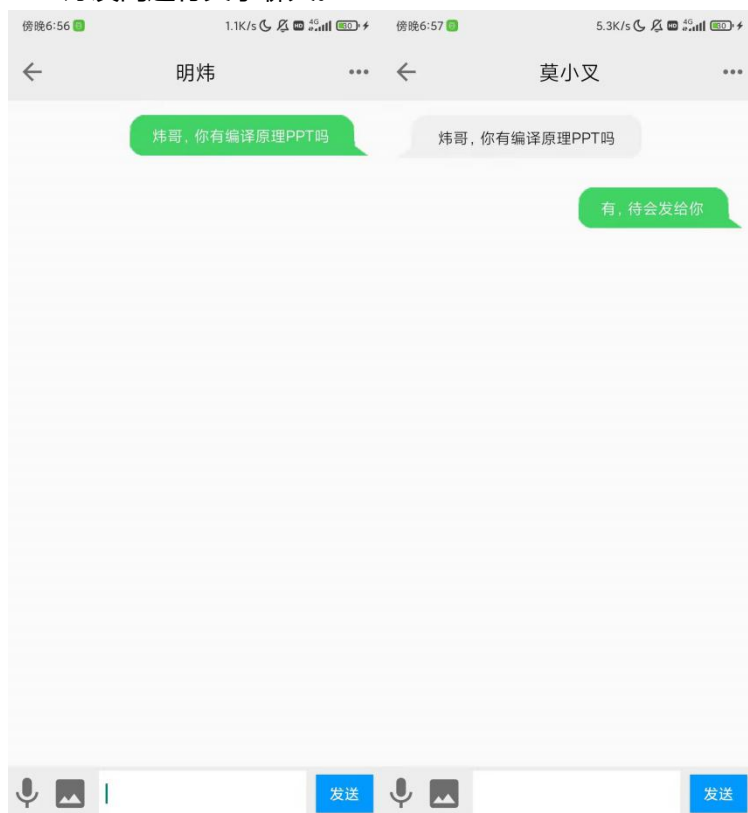
A screenshot of a mobile application's login screen. At the top, the status bar shows the time as 4:51 PM, a data speed of 249K/s, and a 49% battery level. The main content area features a yellow cat emoji as a profile picture. Below it, there is a text input field labeled 'id:' containing the value '777'. Underneath the ID field is a password input field labeled '密码:' with seven dots representing the password. At the bottom of the form, there are two buttons: a green button labeled '登录' (Login) and a white button with a grey border labeled '注册' (Register).

4.2.2. 聊天

登录完成后进入聊天页面可以与好友进行聊天。



好友间进行文字聊天。

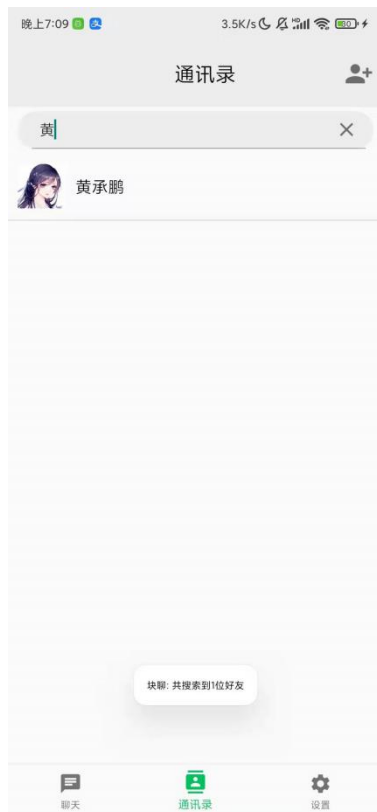


4.2.3. 好友

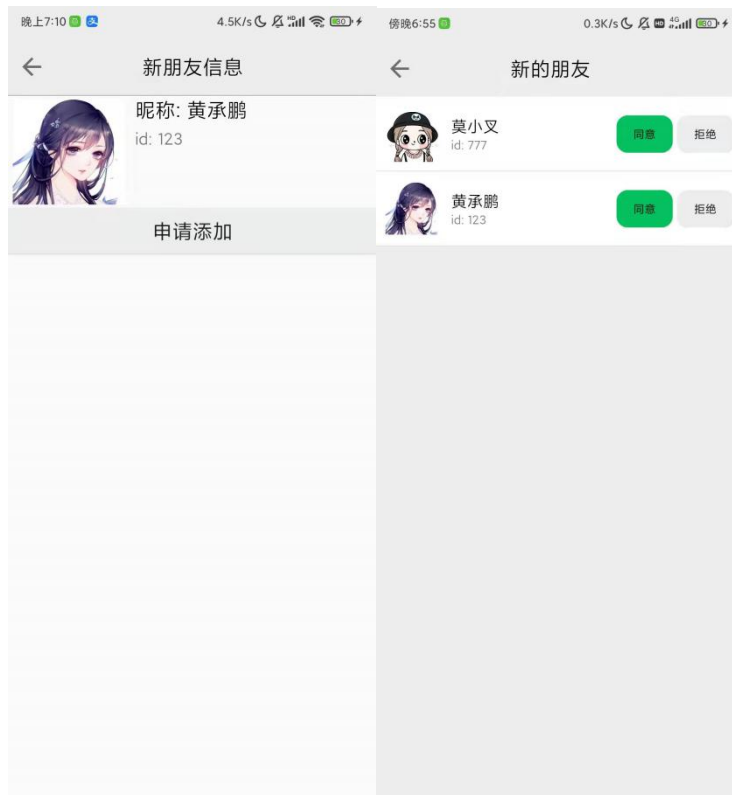
通讯录可以进行查看、搜索、添加好友。



搜索已添加好友。



用户间可以相互添加成为好友。



可以修改好友昵称或删除好友



4. 2. 4. 设置

设置模块可以设置账号昵称、头像、密码等。



修改昵称、设置头像、修改密码。

