**软件项目详细设计**

**（Low Level Design）**

项目名称：喵呜

学校班级：华南师范大学计算机学院15级软工

小组成员：詹萍 陈玉淋 李洁莹 庞倩婷

目录

[一、 引言 3](#_Toc499740229)

[1、编写目的 3](#_Toc499740230)

[2、背景 3](#_Toc499740231)

[3、定义 3](#_Toc499740232)

[4、参考资料 4](#_Toc499740233)

[二、 程序系统的组织结构 4](#_Toc499740234)

[三、 模块“猫圈”设计说明 6](#_Toc499740235)

[1、程序功能描述 6](#_Toc499740236)

[2、性能 7](#_Toc499740237)

[3、算法 7](#_Toc499740238)

[4、流程逻辑 11](#_Toc499740239)

[5、存储分配 15](#_Toc499740240)

[6、注释设计 16](#_Toc499740241)

[7、限制条件 16](#_Toc499740242)

[8、测试计划 16](#_Toc499740243)

[9、尚未解决的问题 17](#_Toc499740244)

# 引言

## 1、编写目的

“喵呜”的详细设计是系统设计第二个阶段的任务，这个阶段的主要是在总体设计的基础上，对总体设计中的功能模块进行详细的过程描述，设计处功能模块的内部细节，包括该功能模块的算法和详细数据结构，用伪代码来表现程序设计的思想，为编写源代码提供必要的说明。

目前，总统设计已经解决了软件系统总体结构设计的问题，包括整个软件系统的结构、模块划分、模块功能和模块间的联系等。

详细设计则要解决如何实现各个模块的内部功能，即模块设计。具体来说，详细设计就是要为已经产生的各个模块设计详细的算法。但这并不等同于系统实现阶段用具体的语言编码，它只是对实现细节作精确的描述，可以采用伪代码等形式来表达，这样编码阶段就可以将详细设计中对功能实现的描述，直接转化为程序设计语言书写的程序。

## 2、背景

由于流浪猫数量的增加，流浪病猫治疗、猫咪绝育、猫咪喂养及猫咪收养问题日益严峻，但目前爱猫协会缺乏统一的管理机制。我们希望通过这个app为爱猫协会及爱猫人士完成猫咪喂养，募捐，救助猫咪，寻找猫咪失主以及猫咪收养等工作打造一个统一且便捷的平台，并且借由此平台吸引更多人士对猫咪的关注与关爱。

## 3、定义

1. 猫圈——类似于朋友圈、微博等，可供用户发关于猫咪动态的平台。
2. 小鱼干——类似于朋友圈点赞功能。
3. Editview——android控件（可编辑文本）。
4. Imageview——android控件（展示图片）。

## 4、参考资料

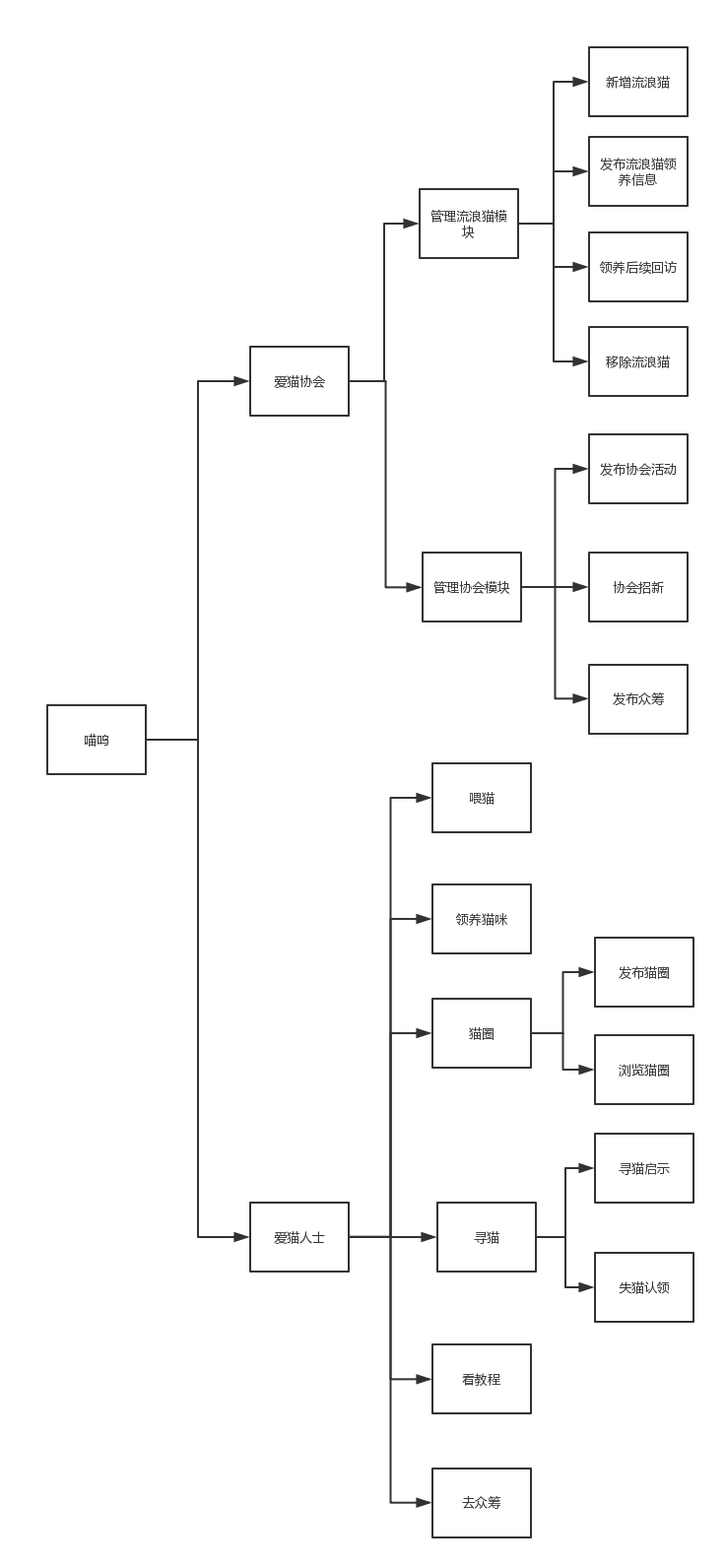
[1]张海藩，牟永敏. 软件工程导论（第6版）.清华大学出版社，2013

[2]Leszek A.Maciaszek，Bruc Lee Liong. 实用软件工程. 机械工业出版社

[3]《详细设计说明书实例\_图书管理系统》

[4]周苏，彭彬，张咏，王文. 软件体系结构与设计. 清华大学出版社，2013

# 程序系统的组织结构



# 模块“猫圈”设计说明

## 1、程序功能描述

**（1）功能描述**

猫圈用于用户发布关于猫的消息，类似微信的朋友圈，空间的说说，微博等。用户也可以给别的猫圈评论，点赞送小鱼干，用户收到的小鱼干可积累积分获得相应奖励。

**（2）界面效果**



**（3）页面元素**

查询列表：用户名、用户头像、详细内容、图片、小鱼干数量、评论数量

查询条件：推荐、最新、附近、关注、搜索（模糊）

记录排序字段：按照发布猫圈的日期降序排列

界面按钮和提示信息说明：

|  |  |
| --- | --- |
| 按钮名称 | 功能说明 |
| 搜索 | 根据用户名或猫圈关键字搜索某一用户发布的猫圈 |
| 发布 | 发布新的猫圈 |
| 推荐 | 系统根据用户个人喜好推荐的猫圈 |
| 最新 | 全部用户发布的最新的猫圈 |
| 附近 | 附近（发布时有显示位置信息的）用户发布的猫圈 |
| 关注 | 用户关注的其他用户发布的猫圈 |
| 小鱼干 | 用户可以给喜欢的猫圈点赞送小鱼干，小鱼干计入该条猫圈的发布者用户账户中 |
| 评论 | 用户可以在其他用户的猫圈下评论 |

## 2、性能

1. 下拉刷新，加载最新的新猫圈、小鱼干、评论；
2. 更新时间为一分钟，每一分钟加载最新的新猫圈、小鱼干、评论。

## 3、算法

**（1）“猫圈”模块处理主要分为两部分**

①发布猫圈

用户需要在发布猫圈界面输入猫圈相关文字信息和图片信息，点击“发布”按钮之后，数据将提交到发布猫圈数据处理页面。取得文字和图片信息后，首先进行输入项检验，若发现输入内容为空，返回并显示发布失败信息。如果输入内容非空，则将信息写入数据库，若写入成功，跳回猫圈界面，若失败，则提示发布失败，需要重新进行发布。

具体伪代码如下：

输入：文字和图片信息

输出：猫圈信息

1.从editview和imageview中获取文字和图片信息；

2.检查是否非空，若为空，则返回输入界面；

3.根据用户id在找到该用户猫圈信息在数据库的位置，将新一条猫圈信息写入数据库，并且获取系统时间，一并写入。若写入失败，返回输入界面；

4.返回猫圈界面，显示发布的猫圈内容

②浏览猫圈

用户进入猫圈界面之后，选择要浏览的信息类型，类型分为：推荐、最新、附近和关注。推荐类型为根据用户平常浏览猫圈信息的兴趣，向用户推荐其感兴趣的猫圈信息；最新为全部猫圈信息按时间排序显示；附近为根据用户所在地理位置，向用户推荐与其地理位置相近的猫圈信息；关注为向用户推荐其关注的人或组织的猫圈信息，并按时间排序显示。系统再获取用户所选的猫圈类型之后，从数据库中挑出对应的猫圈信息，并且显示。

具体伪代码如下：

输入：猫圈类型

输出：该猫圈类型下的猫圈信息

1.获取用户所选猫圈类型；

2.根据类型从数据库中选取相关信息。若失败，则返回；

3.将信息写入显示界面对应的控件，并且按界面要求显示。

**（2）具体代码详情如下（以关注为例）**

public class cat\_circle\_focus extends Activity {

private ListView mListView;

//需要适配的数据

private String[] names={从数据库取出关注人或组织名字，并且按时间排序};

//需要适配的头像图片

private int[] head\_photo={从数据库取出关注人或组织头像，并且按时间排序}

//需要适配的猫圈语句

private String[] content={从数据库取出关注人或组织猫圈语句，并且按时间排序};

//需要适配的照片

private int[] photo\_1={数据库取出关注人或组织猫圈照片，并且按时间排序 };

//需要适配的照片2

private int[] photo\_2={数据库取出关注人或组织猫圈语句，并且按时间排序};

@Override

protected void onCreate(@Nullable Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.cat\_circle\_list);

//初始化ListView控件

mListView=(ListView)findViewById(R.id.cat\_circle\_lv);

//创建一个Adapter的实例

MyBaseAdapter mAdapter =new MyBaseAdapter();

//设置Adapter

mListView.setAdapter(mAdapter);

}

class MyBaseAdapter extends BaseAdapter {

//得到item的总数

@Override

public int getCount() {

//返回ListView Item条目的总数

return names.length;

}

//得到item代表的对象

@Override

public Object getItem(int position) {

return names[position];

}

//得到item的id

@Override

public long getItemId(int position) {

return position;

}

@Override

public View getView(int position, View convertView, ViewGroup viewGroup) {

//将list\_item.xml文件找出来并转换成vi对象

View view=View.inflate(cat\_circle\_focus.this, R.layout.cat\_circle\_list\_item,null);

//为猫圈加名字

TextView tv\_name =(TextView) view.findViewById(R.id.cat\_circle\_tv\_name);

tv\_name.setText(names[position]);

//为猫圈加头像

ImageView imageView =(ImageView) view.findViewById(R.id.cat\_circle\_item\_image);

imageView.setBackgroundResource(head\_photo[position]);

//为猫圈加内容

TextView tv\_content =(TextView) view.findViewById(R.id.cat\_circle\_tv\_content);

tv\_content.setText(content[position]);

//为猫圈加照片1

ImageView imageView1=(ImageView)view.findViewById(R.id.cat\_circle\_iv\_photo1);

imageView1.setBackgroundResource(photo\_1[position]);

//为猫圈加照片2

ImageView imageView2=(ImageView)view.findViewById(R.id.cat\_circle\_iv\_photo2);

imageView2.setBackgroundResource(photo\_2[position]);

return view;

}

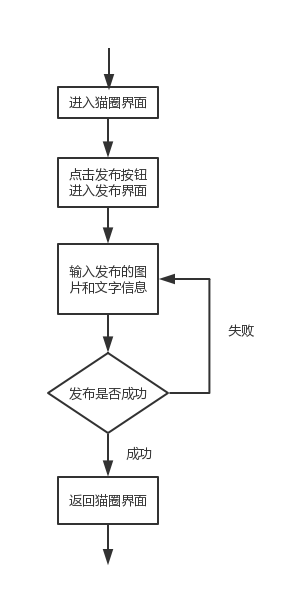
}

}

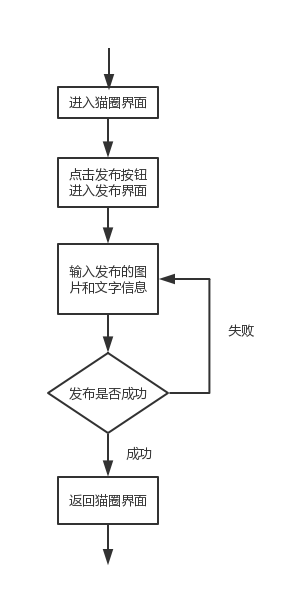
## 4、流程逻辑

**（1）发布猫圈**

用户操作流程图：

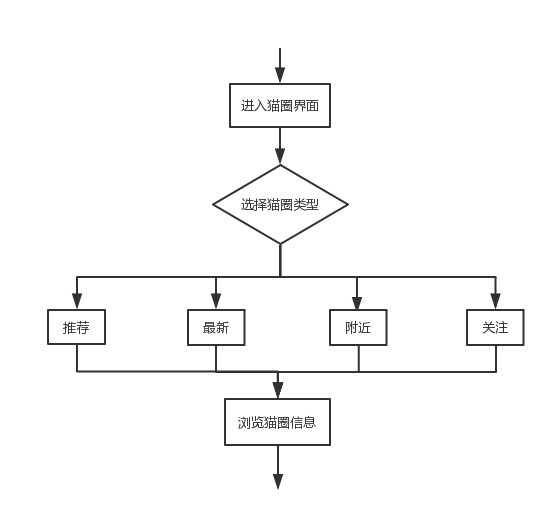


功能实现流程图：

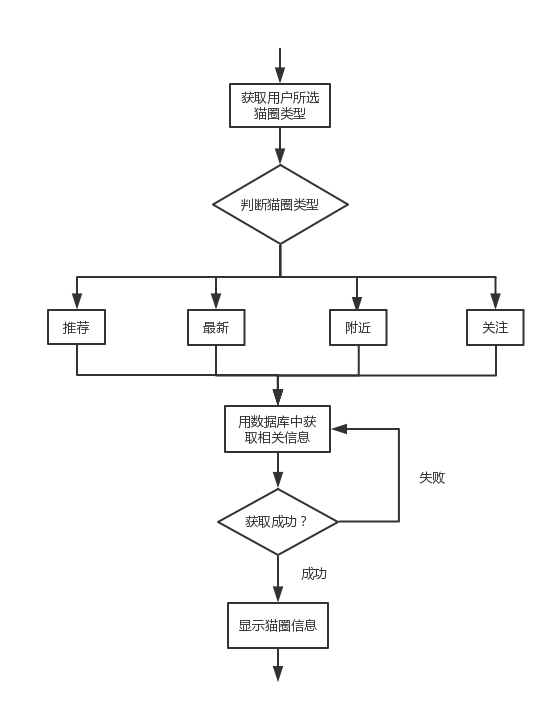


**（2）浏览猫圈：**

用户操作路程图：



功能实现流程图：



## 5、存储分配

表名：cat\_circle\_info

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 类型 | 功能说明 | 备注 |
| Cat\_circle\_id | Int 4 | 猫圈id |  |
| Cat\_circle\_belong | Int 4 | 猫圈所属用户或组织id |  |
| Cat\_circle\_time | Int 4 | 发布猫圈的时间 |  |
| Cat\_circle\_photo | Varchar（20） | 猫圈照片位置 |  |
| Cat\_circle\_content | Varchar（200） | 猫圈文字 |  |
| Cat\_circle\_report | Boolean | 猫圈举报 |  |
| Cat\_circle\_favour | Int 4 | 猫圈获得的赞数量 |  |

## 6、注释设计

（1）功能注释：对某一模块的代码块进行说明，帮助程序员更好的理解代码。

（2）类注释：指明每个类是实现哪方面功能的。

（3）方法注释：说明方法是实现什么功能的。

（4）属性注释：标明属性代表的是什么。

## 7、限制条件

说明本模块运行中所受到的限制条件。详细描述系统所受的内部和外部条件的约束和限制说明。包括业务和技术方面的条件与限制以及进度、管理等方面的限制。

## 8、测试计划

（1）技术要求

熟悉android studio单元测试；

（2）输入数据与预期结果规定

①发布猫圈测试

1. 输入：文字和图片信息为空；

预期结果：返回输入界面，对话框提示用户未输入信息；

1. 输入：文字和图片信息非空

预期结果：写入成功查看数据库/写入失败，返回输入界面；

1. 浏览猫圈

1.输入：猫圈类型——推荐

预期结果：输出该猫圈类型下的猫圈信息/失败返回

2.输入：猫圈类型——最新

预期结果：输出该猫圈类型下的猫圈信息/失败返回

3.输入：猫圈类型——附近

预期结果：输出该猫圈类型下的猫圈信息/失败返回

4.输入：猫圈类型——关注

预期结果：输出该猫圈类型下的猫圈信息/失败返回

## 9、尚未解决的问题

使用数据库存储数据，将猫圈的数据与后台数据库连接。