

**失物招领APP（寻找宝贝）**

题 目 高级软件工程

专 业 软件工程

班 级 软件14-2

姓 名 孙晓聪

学 号 201417020219

指 导 教 师 李来春

2017年6月4日

**目录**

[摘要 IV](#_Toc26975)

[Abstract V](#_Toc29768)

[第一章 绪论 1](#_Toc1169)

[1.1 课题背景 1](#_Toc7315)

[1.2 研究状况和发展趋势 1](#_Toc9766)

[1.2.1 国外研究现状 1](#_Toc20502)

[1.2.2 国内研究现状 1](#_Toc3069)

[1.3 课题意义 1](#_Toc16910)

[1.4 指导思想 1](#_Toc23996)

[第二章 方案论证 2](#_Toc8063)

[2.1 开发方案 2](#_Toc28798)

[2.1.1 SSM+Android 方案 2](#_Toc31899)

[2.1.2 SSS+Android 方案 2](#_Toc21801)

[2.2 可行性分析 2](#_Toc9791)

[2.2.1 经济可行性 2](#_Toc28269)

[2.2.2 技术可行性 2](#_Toc16508)

[2.2.3 法律可行性 2](#_Toc29622)

[2.2.4 操作可行性 2](#_Toc8775)

[2.3 结论 3](#_Toc12428)

[第三章 需求分析 4](#_Toc8129)

[3.1 需求描述 4](#_Toc3142)

[3.2 系统描述 5](#_Toc16317)

[3.3 数据流分析 6](#_Toc1013)

[3.3.1 顶层数据流图 6](#_Toc32360)

[3.3.2 一层数据流图 6](#_Toc20713)

[3.3.3 二层数据流图 7](#_Toc3651)

[3.4 数据分析 7](#_Toc2128)

[3.4.1 实体关系分析 7](#_Toc20937)

[第四章 过程论述 9](#_Toc9391)

[4.1 概要设计 9](#_Toc23420)

[4.1.1 架构设计 9](#_Toc12361)

[4.1.2 拓扑结构设计 9](#_Toc28466)

[4.1.3 功能模块设计 10](#_Toc16314)

[4.1.4 功能活动设计 10](#_Toc5836)

[4.1.5 界面设计 11](#_Toc9794)

[4.1.6 系统包设计 13](#_Toc11726)

[4.1.7 系统类设计 14](#_Toc22935)

[4.2 详细设计 15](#_Toc8800)

[4.2.1 子模块设计 15](#_Toc26948)

[4.2.2 数据库设计 17](#_Toc24523)

[4.2.3 代码实现 17](#_Toc19919)

[第五章 结果分析 18](#_Toc13679)

[5.1 白盒测试 18](#_Toc13884)

[5.2 黑盒测试 18](#_Toc26299)

[第六章 结论 19](#_Toc28344)

[第七章 致谢 20](#_Toc13247)

[第八章 参考文献 21](#_Toc4452)

[第九章 附录 22](#_Toc26427)

# 摘要

设计这个失物招领App是出于两方面的原因，第一是Android 嵌入式开发实验的需要，第二是前段时间在3304上课的时候，上一节课上课的同学总是落下U盘，手机和钥匙之类的东西，然后老师还不知道还给谁。这时候就萌生了做一个这样的工具App，可以将丢失物品的信息传递给失主。

本失物招领App命名为“寻找宝贝”，英文名为FindBaby。分为App客户端和Web服务端。App客户端实现Android平台，Web服务端采用SSS(Spring MVC,Spring,Spring JDBC)框架集合开发基于服务的Web(SOA)，并且采用Restful Api。

关键词：失物招领，FindBaby， Android，SSS，Restful

# Abstract

The design of this lost property App is based on two reasons, the first is the need for Android embedded development experiments, the second is some time ago in the 3304 class, the last class students always fall U disk, mobile phone, Key and something like that, and then the teacher did not know who to return. This time on the initiation of such a tool App, you can send the information of the items to the owner.

The lost property App named "looking for baby", English called FindBaby. And including App client and Web server. App client implements the Android platform, the Web server uses the SSS (Spring MVC, Spring, Spring JDBC) framework to develop a Service-Oriented Architecture (SOA)and uses Restful API.

Keywords: lost property, FindBaby, Android, SSS, Restful

# 第一章 绪论

## 1.1 课题背景

Android很火，嵌入式很火，移动开发很火，所以我们开了嵌入式开发课程。

## 1.2 研究状况和发展趋势

### 1.2.1 国外研究现状

国外有Google公司和苹果公司牵头移动领域，占据绝对优势，有雄厚的研究实力，掌握移动领域的发展。

### 1.2.2 国内研究现状

虽然国外的优势很明显，但是我们国内长城防火墙，阻挡了很大一部分科技的冲击。给国内的民族开发商以缓冲的机会。

## 1.3 课题意义

本课题旨在设计实现一个App，可以将失物信息传递给失主，方便失主找回。所以在开发个过程中，可以锻炼Android App开发和Web 服务开发的结合。

## 1.4 指导思想

根据软件工程思想和Java 面向对象思想来设计指导。

# 第二章 方案论证

## 2.1 开发方案

### 2.1.1 SSM+Android 方案

SSM是Spring MVC，Spring，MyBatis的缩写，是当下比较流行的JavaWeb一站式开发框架。其中Spring MVC 是Spring生态圈中的一个开发框架，旨在替换Struts2等MVC框架。Spring 是Java 一个框架，旨在简化Java程序的开发，Spring 框架在Java开发中占据非常重要的地位。MyBatis是一个ORM及对象关系映射框架，旨在将领域对象映射到关系型数据库。Android是一个移动操作系统平台，占据了全球绝大部分的移动市场。

### 2.1.2 SSS+Android 方案

SSS是Spring MVC，Spring，Spring JDBC的缩写，是一站式Spring很好的体现。Spring MVC和Spring在上面已经介绍，Spring JDBC 也是Spring 生态圈中的一个子框架，旨在简化JDBC的开发，将JDBC中很多模板代码封装，以方便重用。

## 2.2 可行性分析

### 2.2.1 经济可行性

开发这个App用的都是开源软件，无需任何经济需要。

### 2.2.2 技术可行性

因为用的都是开源且比较流行的框架，能够在官网或者社区得到技术支持。但是基于SSM+Android的框架过于庞大，相反SSS+Android的框架跟适合可以这样的小团队。

### 2.2.3 法律可行性

都是开源软件无版权问题。

### 2.2.4 操作可行性

开发的App是基于Android 移动操作系统，交互友好。

## 2.3 结论

综上所述采用SSS+Android方案更合适，同时无其他顾虑。

# 需求分析

## 3.1 需求描述

失物招领的App的需求是失物拾取者以下简称“好心人”，将失物以下简称“宝贝”的信息传送给失主同时附带自己的电话号码，电话号码需要失主答对问题过后才会获得，以便失主通过电话来获取自己的宝贝。

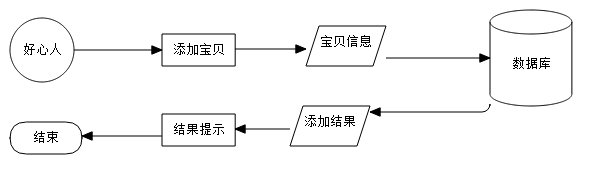


图 3-1 添加宝贝流程图

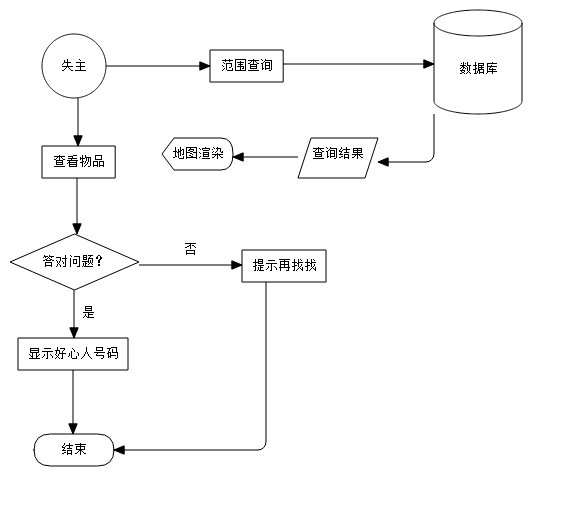


图 3-2 寻找宝贝流程图

## 3.2 系统描述

此系统包含两种角色好心人和失主，好心人包含添加宝贝用例，失主包含寻找宝贝用例和回答问题用例。

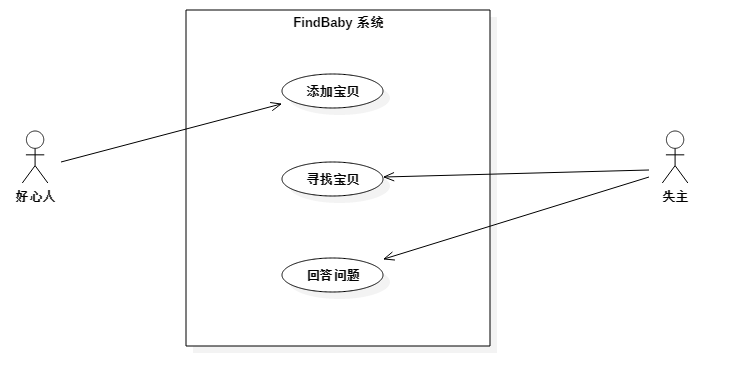


图 3-3 系统用例图

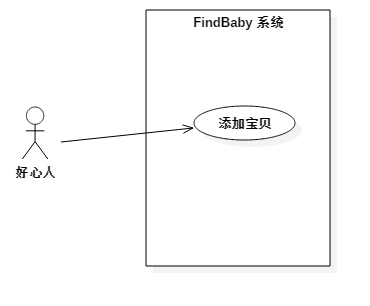


图 3-4 好心人用例图

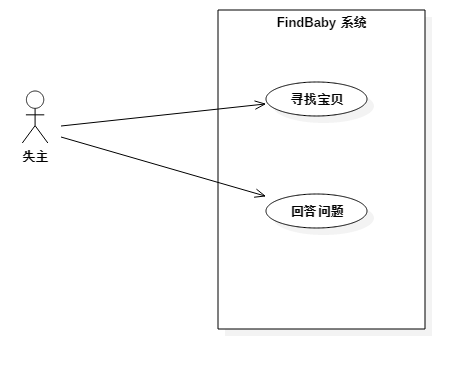


图 3-5 失主用例图

## 3.3 数据流分析

### 3.3.1 顶层数据流图

顶层数据流顺序是好心人-FindBaby系统-失主，数据流图如下：



图 3-5 顶层数据流图

### 3.3.2 一层数据流图

一层数据流图是从好心人-添加服务-数据库-查询服务-失主，一层数据流图如下：



图 3-6 一层数据流图

### 3.3.3 二层数据流图

二层数据流图从好心人-添加控制器-添加服务-添加持久化层-数据库-查询持久化层-查询服务-查询控制器-失主，二层流程图如下：

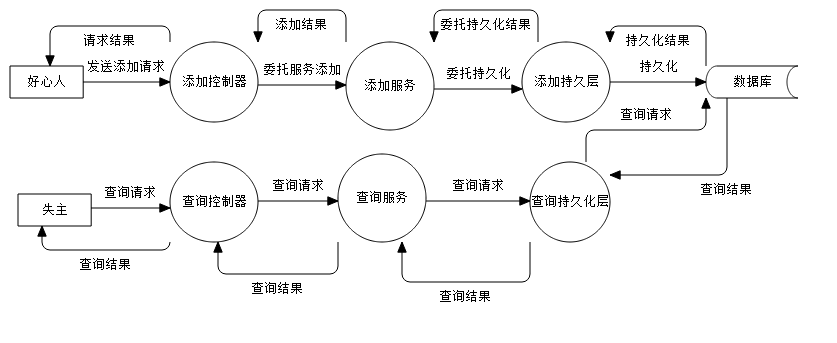


图 3-7 二层数据流图

## 3.4 数据分析

### 3.4.1 实体关系分析

根据以上分析，提炼出来了两个实体：Baby和Log，并绘制了如下E-R图：

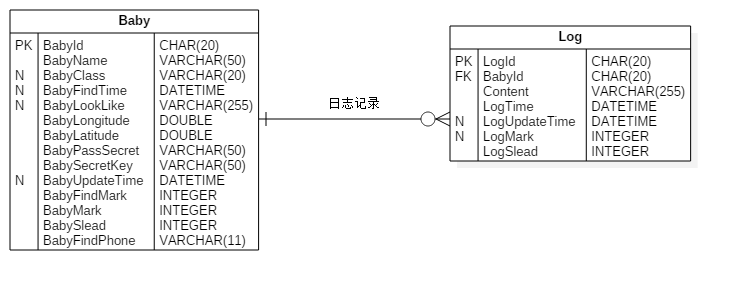


图 3-8 E-R图

# 过程论述

## 4.1 概要设计

### 4.1.1 架构设计

本失物招领系统采用C/S架构设计，及客户端服务器架构。客服端实现Android平台，服务端采用JavaWeb，他们之间的通信基于Http。

具体流程为App 向服务器发送Get/Post请求，服务器处理并响应请求，系统架构图如下：

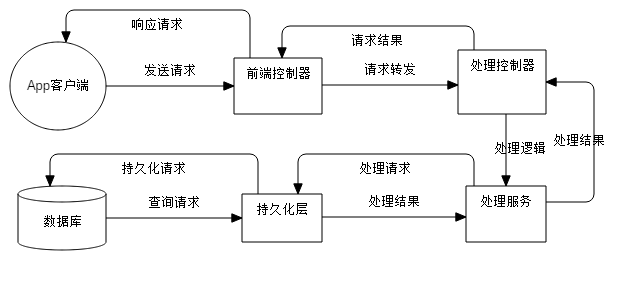


图 4-1 系统架构图

### 4.1.2 拓扑结构设计

系统拓扑结构为客户端App，服务器端JavaWeb服务器和数据库，具体的拓扑结构如下：

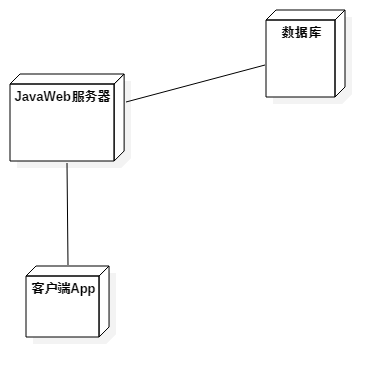


图 4-2 系统拓扑结构图

### 4.1.3 功能模块设计

根据实际情况本失物招领系统分为3个模块，添加宝贝，寻找宝贝，回答问题。具体的系统功能模块图如下：

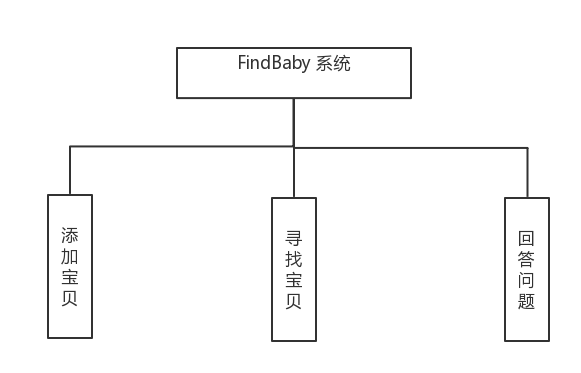


图 4-3 系统功能模块图

### 4.1.4 功能活动设计

在本系统内逻辑上是有两种角色及好心人和失主，但是出于工作量的原因只是在逻辑上区分了两种角色，在程序中只有用户一种角色以下简称用户。根据子模块和业务流程绘制了如下的活动图：

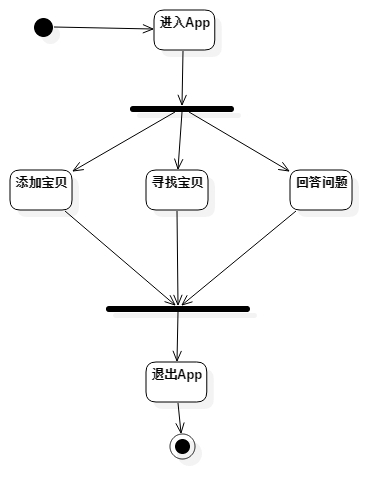


图 4-4 用户活动图

### 4.1.5 界面设计

界面设计主要是Android这一块，用户主要是跟Android App进行交互，一下是设计的主要交互界面：

1. App 主界面



图 4-5 App主界面

1. 添加宝贝界面



图 4-6 App添加宝贝界面

1. 寻找宝贝界面



图 4-7 App寻找宝贝界面

1. 寻找结果界面



图 4-8 App寻找结果界面

1. 回答问题界面



图 4-9 App回答问题界面

### 4.1.6 系统包设计

通过对系统的分析，系统客户端和服务端分别属于不同的项目，所以对其包进行单独设计。

1. Android 项目主要包：

A) activity包主要存放活动界面

B) model存放领域对象

C) util 存放工具类

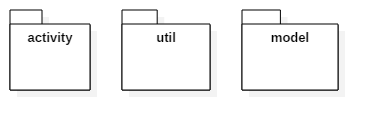


图 4-10 Android项目 包图

1. JavaWeb 项目主要包：
2. model 存放领域对象
3. config 存放配置文件
4. dao 数据访问层
5. service 业务逻辑层
6. util 工具类
7. web 控制器类

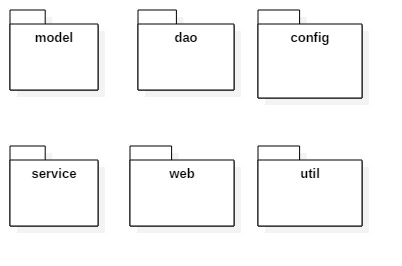


图 4-11 JavaWeb项目 包图

### 4.1.7 系统类设计

根据系统的设计要求对系统的服务端的主要类绘制了类图：

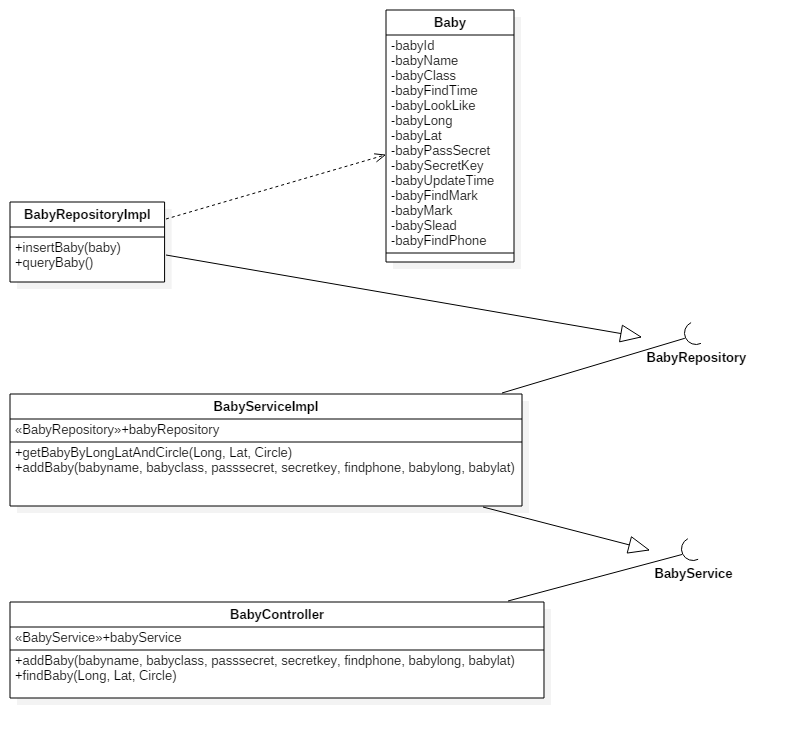
、

图 4-12 系统服务端类图

## 4.2 详细设计

### 4.2.1 子模块设计

在上面的模块这几种，将本系统分为三个子模块添加宝贝，寻找宝贝和回答问题。下面分别对这三个子模块进行设计。

1. 添加宝贝子模块

添加宝贝子模块的功能是将好心人找的的丢失物品信息添加到数据库，方便失主寻找，根据业务流程绘制了如下时序图：

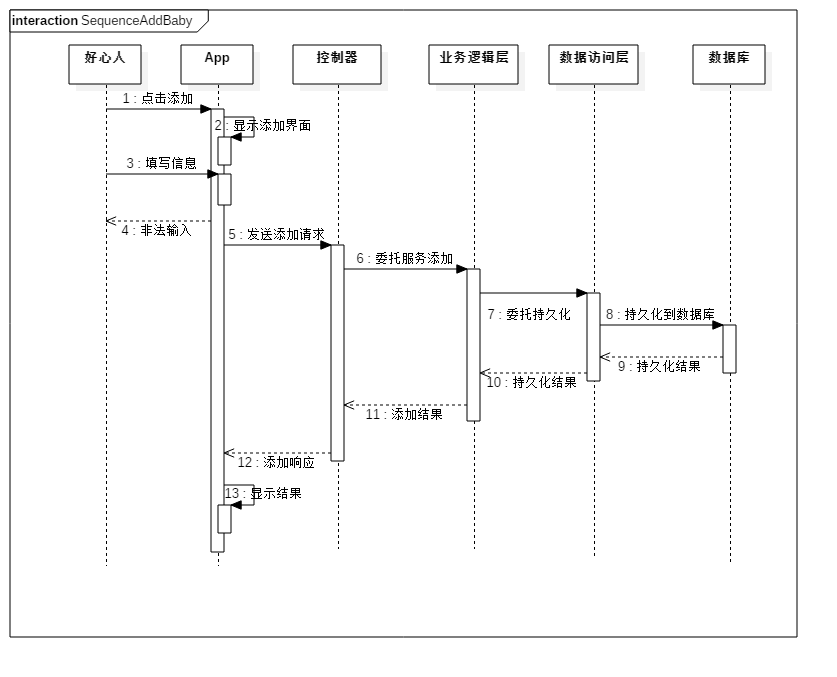


图 4-13 添加宝贝时序图

1. 寻找宝贝子模块

寻找宝贝模块是失主在App填写好搜索半径过后，提交给服务器由服务器代为搜索，搜素搜完成后服务器将结果返回给App显示给失主看，根据流程绘制了如下的时序图：

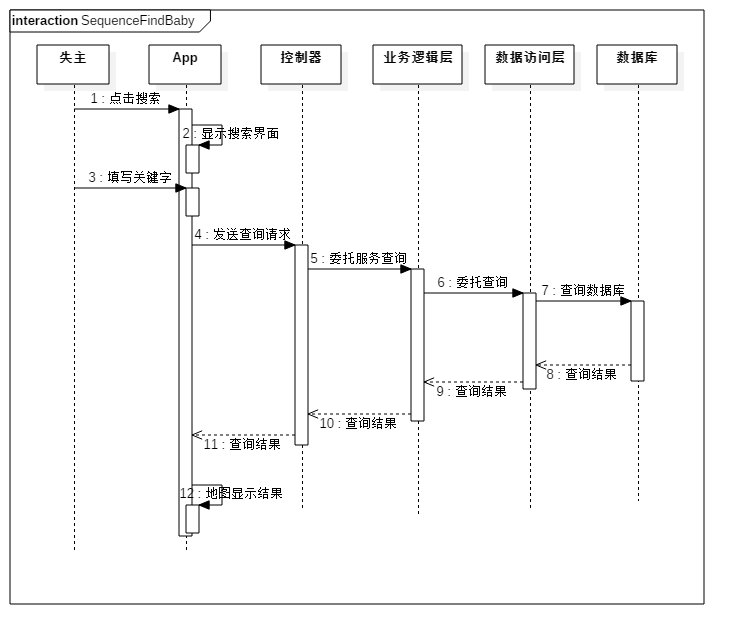


图 4-14 寻找宝贝时序图

1. 回答问题子模块

回答问题子模块是失主在地图上点击某个地图选点后，弹出问题，在失主回答完毕问题后显示回答结果。如果正确显示好心人手机号码否则显示提示信息，时序图如下：

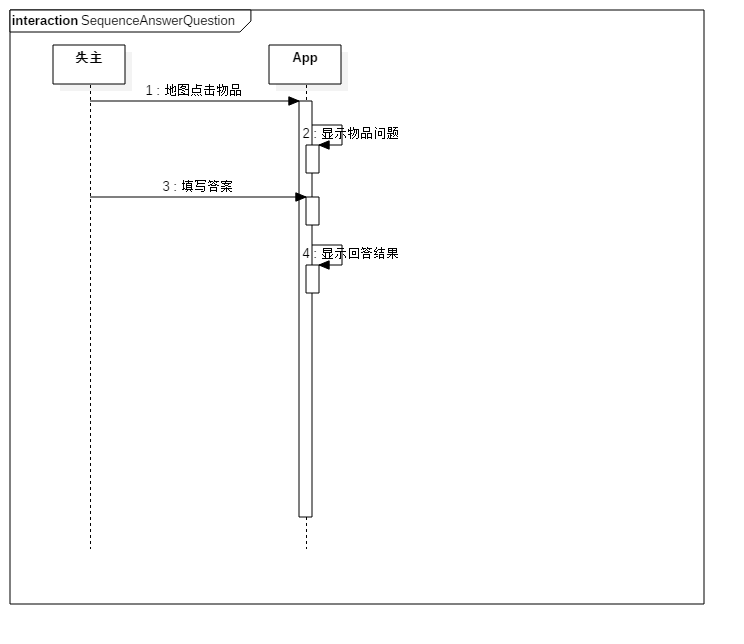


图 4-15 回答问题时序图

### 4.2.2 数据库设计

数据库是基于MySql开源数据库来设计数据库表的，总共有两个表，Baby实体和Log实体，具体设计如下：

1. Baby 实体表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名称** | **字段名** | **类型** | **长度** | **主键** | **非空** |
| 编号 | BabyId | CHAR | 20 | YES | YES |
| 名称 | BabyName | VARCHAR | 50 | NO | YES |
| 类别 | BabyClass | VARCHAR | 20 | NO | NO |
| 找到时间 | BabyFindTime | DATETIME | - | NO | NO |
| 外观 | BabyLookLike | VARCHAR | 255 | NO | NO |
| 经度 | BabyLongtitude | DOUBLE | - | NO | YES |
| 维度 | BabyLatitude | DOUBLE | - | NO | YES |
| 问题 | BabyPassSecret | VARCHAR | 50 | NO | YES |
| 答案 | BabySecretKey | VARCHAR | 50 | NO | YES |
| 更新时间 | BabyUpdateTime | DATETIME | - | NO | NO |
| 还回标记 | BabyFindMark | INTEGER | - | NO | YES |
| 标记 | BabyMark | INTEGER | - | NO | YES |
| 是否删除 | BabySlead | INTEGER | - | NO | YES |
| 电话号码 | BabyFindPhone | VARCHAR | 11 | NO | YES |

1. Log 实体表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **名称** | **字段名** | **类型** | **长度** | **主键** | **非空** |
| 日志编号 | LogId | CHAR | 20 | YES | YES |
| 宝贝编号 | BabyId | CHAR | 20 | NO | YES |
| 操作内容 | Content | VARCHAR | 255 | NO | YES |
| 操作时间 | LogTime | DATETIME | - | NO | YES |
| 修改时间 | LogUpdateTime | DATETIME | - | NO | NO |
| 标记 | LogMark | INTEGER | - | NO | NO |
| 是否删除 | LogSlead | INTEGER | - | NO | YES |

### 4.2.3 代码实现

关键代码主要是服务器端的代码和App的代码实现，篇幅有限就不在这儿贴代码了，代码已经上传到GitHub可以在线下载，下面附上地址：

1. 整个项目地址

<https://github.com/brainysoon/FindBaby>

1. 服务器项目地址

<https://github.com/brainysoon/FindBaby/tree/master/BaseFind>

1. Android端项目地址

<https://github.com/brainysoon/FindBaby/tree/master/PopFind>

# 结果分析

## 5.1 白盒测试

在编写代码的时候分别对数据访问层，业务逻辑层和视图展现层进行了单元测试，编写了测试代码，测试代码编写是基于Java测试框架JUnit实现。

## 5.2 黑盒测试

写好App和服务器端后，将服务器端部署在了腾讯云的服务器上，然后将App生成Apk后发布。让组员分别对App一阵玩，勉强能够挺下去不至于崩溃。

# 结论

通过这个实验，有机会将Android App和Java Web 结合起来，实现了一整套服务的整合，从中学习到了不少。虽然这个App很小但是麻雀虽小五脏俱全。

# 致谢

最后感谢李老师的嵌入式开发课程实验和张老师的高级软件工程实验，能让我有机会将文档与代码实现结合起来。

# 参考文献

【1】高金昌，张明星著 著《Android底层驱动分析和移植》 清华大学出版社 2015-07

【2】方木云，刘辉 著 《高级软件工程》 清华大学出版社 2011-06

# 附录

贴上项目仓库地址：代码实现，UML文档，论文都在仓库里面

<https://github.com/brainysoon/FindBaby>