

代 号 10701

分 类 号 TP311.5

U D C

学 号 10072032

密 级 公开

编 号

题（中、英文）目 基于智能医疗平台的“医键通”系统设计与实现

Design and Implementation of easy calling System

Base On Smart Medical Platform

作 者 姓 名 南敏庚 学校指导教师姓名职称 武波 教授

工 程 领 域 软件工程 企业指导教师姓名职称 潘阳 高工

提交论文日期 二〇一一年六月

西安电子科技大学 学位论文独创性（或创新性）声明

秉承学校严谨的学风和优良的科学道德，本人声明所呈交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果；也不包含为获得西安电子科技大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中做了明确的说明并表示了谢意。

申请学位论文与资料若有不实之处，本人承担一切的法律责任。

本人签名：_____

日期_____

西安电子科技大学 关于论文使用授权的说明

本人完全了解西安电子科技大学有关保留和使用学位论文的规定，即：研究生在校攻读学位期间论文工作的知识产权单位属西安电子科技大学。学校有权保留送交论文的复印件，允许查阅和借阅论文；学校可以公布论文的全部或部分内容，可以允许采用影印、缩印或其它复制手段保存论文。同时本人保证，毕业后结合学位论文研究课题再撰写的文章一律署名为西安电子科技大学。（保密的论文在解密后遵守此规定）

本学位论文属于公开，适用本授权书。

本人签名：_____

日期_____

导师签名：_____

日期_____

摘 要

随着人口老龄化的来临和人民生活水平的提高，如何利用通讯和软件，提供更加智能和快捷方便的医疗服务和急救能力，已经成为社会主要关注的问题。目前全球大型公司从 IBM 的智慧医疗，到微软的医疗协议解决方案，以及 google 和 Intel 智能家居,都在进行全方位的布局，但如何结合中国国情，开发适合中国老年人使用的智能医疗服务系统，目前还没有比较完整的解决方案。

本文在分析手机和呼叫中心、自动工单流程核对、短信收发、以及如何使用手机短信和呼叫中心进行老年人远程呼叫服务等相关技术的基础上，借鉴 SaaS 和 SOA 方面的很多经验和思想，设计了老年人远程呼叫系统的终端和软件，给出了实现该系统的通讯协议和软件控制流程。包括终端和服务器之间的通讯，服务器和坐席之间的通讯，包括多次重复呼叫，呼叫中途更改，短信呼叫变更为语音呼叫，异常情况处理，处理情况跟踪，处理流程自动核实；语音报时，终端时间自动核对，信号强弱语音提示，电量语音提示。

本系统可以应用于社区医疗，农村医疗，老年护理，养老院，残疾人保障中心，急救中心，家庭，医院等各种场所。

关键词：医疗服务 急救 无线呼叫 老年护理

Abstract

Moves towards the aging and developed country population aging development while the world population, the Chinese population aging is also intensifying. how to use of communication and software technique, to provide more convenient and quick intelligence of the medical services and first aid skills, has become a major concern. The current global major companies from ibm, microsoft the wisdom of the medical treatment protocol solution, google and intelligence, intel home in a comprehensive arrangement, but how to combine the development of china, for the use of medical services system, there were no more complete solutions.

This article analyzed cellular phone and tradition to call a centre and was used as the foundation to it, Combine an automatic work unipole to check, the message receives and dispatches, Propose conceiving of old age call service. and the soa saas of a lot of experience and thinking, the remote calls the device and software, to implement the system of communication protocol and software control. Include terminals and the communication between servers server, and reclining at the communication, including many times repeated calls, call changes, voice call change to SMS, the handling of the track, flow automatically verify; time by voice, Automatically check and fix time , signal intensity prompt ,power prompt by voice.

This system can be applied to the community health care and rural medical and nursing homes and old age, the security and the first aid centers, the family, the hospital. for various.

Keyword: Medical services First aid Wireless calls Elderly care

目 录

| | |
|-----------------------------------|----|
| 第一章 绪论..... | 1 |
| 1.1 论文背景..... | 1 |
| 1.2 国内外产品技术现状分析..... | 1 |
| 1.3 现有系统的技术分析..... | 4 |
| 1.4 本文工作..... | 5 |
| 1.4 章节安排..... | 6 |
| 第二章 医键通系统相关技术介绍..... | 7 |
| 2.1 接入技术..... | 7 |
| 2.1.1 语音业务接入概述..... | 7 |
| 2.1.2 短信服务（SMS）接入概述..... | 9 |
| 2.1.3 GPRS 接入概述..... | 11 |
| 2.2 服务器与电子商务平台连接技术研究..... | 13 |
| 2.2.1 概述..... | 13 |
| 2.2.2 系统逻辑划分..... | 13 |
| 2.2.3 系统协议说明..... | 14 |
| 2.2.4 RIA 和 Silverlight 技术介绍..... | 15 |
| 2.3 系统整体架构技术研究..... | 17 |
| 2.3.1 组网方式..... | 17 |
| 2.3.2 设备终端..... | 18 |
| 2.3.3 短信接入..... | 19 |
| 2.3.4 数据库部分..... | 19 |
| 2.3.5 呼叫中心内部结构图网络规划介绍..... | 19 |
| 2.3.6 UML 概述..... | 20 |
| 2.3.7 RUP 概述..... | 21 |
| 2.3.8 设计模式..... | 24 |
| 2.3.9 SaaS..... | 24 |
| 2.4 本章小结..... | 25 |
| 第三章 系统总体分析与设计..... | 27 |
| 3.1 系统总体设计..... | 27 |
| 3.2 终端的设计主要内容..... | 28 |
| 3.3 系统后台软件设计主要内容..... | 30 |
| 3.4 终端设计方案..... | 32 |
| 3.5 后台软件系统设计方案..... | 32 |

| | |
|-----------------------|----|
| 3.5.1 软件系统功能划分..... | 32 |
| 3.5.2 模块功能描述..... | 32 |
| 3.5.3 性能要求..... | 33 |
| 3.5.4 系统用例图..... | 34 |
| 3.6 SaaS 模式的应用设计..... | 37 |
| 3.7 本章小结..... | 38 |
| 第四章 系统详细设计及实现..... | 39 |
| 4.1 终端的详细设计..... | 39 |
| 4.2 后台模块详细设计和实现..... | 44 |
| 4.3 详细描述概述..... | 45 |
| 4.3.1 类的层次关系..... | 45 |
| 4.3.2 实体类..... | 47 |
| 4.3.3 数据库设计..... | 48 |
| 4.3.4 操作类..... | 52 |
| 4.3.5 界面类..... | 53 |
| 4.4 短信服务器代码分析..... | 54 |
| 4.5 编码实现..... | 57 |
| 4.6 本章小结..... | 58 |
| 第五章 系统测试..... | 59 |
| 5.1 测试环境..... | 59 |
| 5.2 测试方案..... | 60 |
| 5.2.1 功能测试..... | 60 |
| 5.2.2 性能测试..... | 61 |
| 5.2.3 测试结果..... | 61 |
| 5.3 小结..... | 64 |
| 第六章 结束语..... | 65 |
| 6.1 工作总结..... | 65 |
| 6.2 未来展望..... | 65 |
| 致谢..... | 67 |
| 参考文献..... | 69 |
| 附录 A..... | 71 |

第一章 绪论

本章主要阐述本课题的产生背景，国内外对此课题的研究现状，以及对目前此课题的国内产品的分析。最后给出了本文的研究内容、研究重点和章节安排，以便读者能清晰的了解论文的思路。

1.1 论文背景

老年护理是一个我国目前急需解决的问题，伴随着医疗体制改革的不断深化和医疗事业的飞速发展，越来越多的老年人需要快捷、方便地得到医院的各种各样的医疗服务；衡量老年护理的综合水平高低，不再仅仅局限于软、硬件的水平高低，更要看服务水平的综合能力。原有的医疗服务体系是以医院为单位，已不适应现代医疗需求，现在医疗服务逐步转向以预防为主，所以社区医疗服务和老年医疗服务逐渐成为现代医疗需求的主要服务内容，是大多数医疗单位计划或正在做的工作。如何利用先进的信息技术为医疗事业服务，更大程度的提高老年护理的服务质量及内容，是医疗信息化建设中的一个重要着眼点，也是本系统要解决的一个问题。

随着科技水平和医疗水平的进步，老年护理的服务理念也随之发生了相应的改变，树立了以人为本，及时服务的服务意识已经成为公认。为了方便患者使用，提高服务质量，远程医疗呼叫系统已经成为了国内外广泛使用的一种电子设备。呼叫系统发展到现在，已经不再是简单的小范围之间沟通的工具，在实际应用当中既要兼顾到社会的整体设计，又要具有良好的实用性、装饰性。可以说，远程医疗呼叫系统已经成为不可缺少的监护设备，是老年护理现代化的标志。它对于病人和医护人员之间的信息沟通起到了至关重要的作用，能够有效地保证病人及时得到医护人员的看护和医治。

本论文主要采用新的通讯方式和自动工单核对的方法，解决远距离传输问题，以及资源避免无谓的争执。既可以帮助病人快速的呼叫得到服务，也可减轻大医院人满为患，无法照顾社区的患者；同时扩大普通医院的营业范围，尤其是一些民营的医院，可以解决其患者不足的问题；还可分散就医地点和服务场所，隔离交叉干扰。因此远程医疗呼叫系统具有广泛的社会意义与重大的实用价值。

1.2 国内外产品技术现状分析

目前国内主要有以下系统：沈阳铁东区“一键式呼叫”，北京以老年手机为主的 999 急救系统，以社区短距离通讯为主的河南某企业的急救呼叫器。这几个

系统目前都进入实际应用阶段。

据介绍，沈阳铁东区民政局和相关部门合作，制作了一批“一键式呼叫”设备。设备设有“服务”和“医疗”两个键，设备的终端将直接和社区以及就近的医院连接，一旦出现突发状况，按键就能立即联系到相应部门，争取治疗和处置时间。

铁东区民政局已经完成了设备研发和入户人员调查。首批 400 部设备将于近期入户到平常老人家，如果居民需要此项服务，可到属地的社区和街道申报，若情况属实将优先成为首批“一键呼叫”服务的受益者。

在铁东区 75 个社区都设有一站四室，老年人可以在其中体验棋牌、运动、书法、医疗等各方面的服务，在各个街道也都设置了养老服务中心。目前铁东区投资百万元建立的养老服务中心有两个，小型的养老服务机构 28 家，早八晚五专门针对社区内的老年人免费开放，部分社区还提供 24 小时的全天候服务。这些养老服务机构全部设有床位和专职的护理人员，养老服务中心也成了真正意义上的“托老所”，老年人可在其中享受到专人服务，足不出户就有“贴身小护士”。

北京 999 急救系统主要是在现在的手机基础上增加了一个比较大的红色按钮，在老年人遇到紧急情况可以按下后，直接拨打 999 服务。

河南社区急救呼叫器在老年人需要帮助时可按下按钮，在社区放置一个接受器，社区服务人员可直到那个老人需要服务。

以上系统实际上都是在原有存在系统上的一个简单的升级，无法满足老年服务的真正要求，携带不方便，使用不灵活，或者服务范围有限。

目前美国已经有一些老年人特别使用的呼叫设备，像一个怀表，携带比较方便，上面有几个紧急按钮，在需要时按下，服务中心就接到求救信号，从而可提供相应服务，与国内产品相比，更符合用户的要求，使用范围也比较大。

这种系统借助手机通讯网络，可以作用比较大的范围，是目前比较贴近用户的产品，但是携带和使用还有一定的不方便。

美国合众思壮是全球最大的车载导航系统厂家之一，其通过定位提供的服务正在从汽车行业拓展到其他行业，它的 GPS 腕表手机产品采用 GPS（全球卫星定位系统）技术、GSM(时分多址个人移动通讯系统)技术、GIS(数字化地理信息系统)技术、网络通讯技术、计算机技术等当今国内外高科技技术，采用灵活方便的实施途径，着力解决目前我国大中城市普遍存在的弱势群体（老人、小孩）的监护问题。

其产品组成如下图 1.1 所示

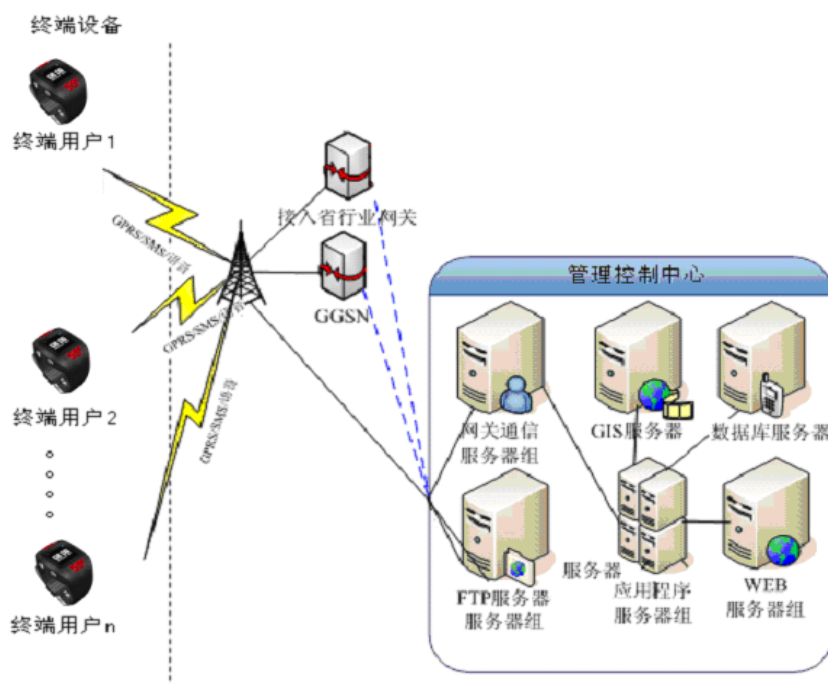


图 1.1 合众思壮的 GPS 腕表手机产品组成

其终端产品如下图 1.2:



图 1.2 终端产品功能示意图

关机：待机状态下，长按开关键关机，关机时后台还有处理，大概二十秒后才完全关机完毕。

SOS 一键求救：开机状态，如果有 SIM 卡：长按 3 秒钟 sos 键，手表先振动一下，并自动发出应急的求救信息，预设的求救信息内容及接收手机号可以自定义设置；如果没有 SIM：长按 3 秒钟 sos 键，则弹出提示是否拨打紧急电话（国内为 112）。

拨出电话：开机状态下长按 call1、call3 或 call2 键 3 秒钟，可拨打预设电话。

设置预设电话：可通过“平台服务”通过网络来设置也可通过监护人手机通过短消息直接设置。目前平台可以设定接入四个电话号，可以打出 3 个电话号和一个 SOS 号码。

接听电话：当有来电时，手表会响铃和振动，并在 LCD 屏幕上显示下列信息作出提醒：显示来电的电话号码，时间。如需要，按 GPS 定位手表上的电话键接听来电。如不想接听，则短按 call3 键或电源键拒接来电。

使用免提听筒：本话机提供通话时自动免提扩音，接插上耳机则自动转至耳机输出模式。

调整音量：待机界面状态下短按侧键的 call1 键可增大铃声音量，短按 call2 键可减小铃声音量。通话状态下按 call1 键可增大通话音量，短按 call2 键可减小通话音量。其功能如图 1.3 所示。

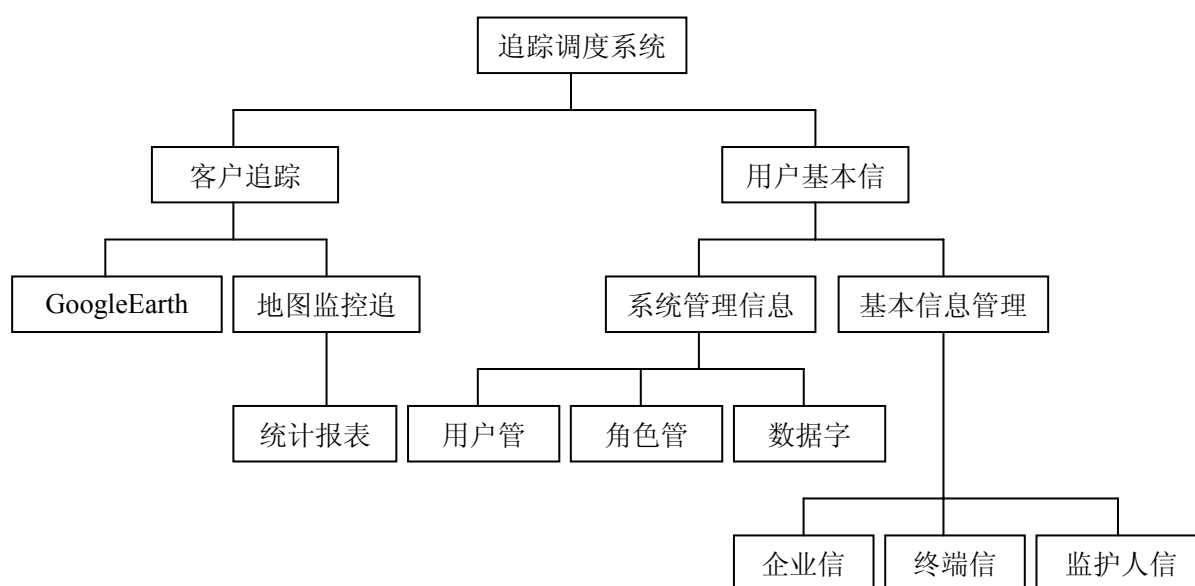


图 1.3 系统功能图

1.3 现有系统的技术分析

从以上情况可以看出，上面相对完善的系统都有一个很大的缺陷，就是真正的业务完全依赖语音业务，使得在有限的坐席数量和电话数量的情况下，无法同时处理更多的用户请求，对用户的服务也没有合适的方法保证用户的请求得到及时满意的处理，同时也无法提供更多更丰富的服务内容。从而无法真正把为老年服务的核心的业务与设备进行有机的结合，只是简单的技术拼凑，没有做为一个完整的系统进行设计。

针对以上不足，本文设计了一套老年人远程呼叫系统的终端和后台软件，给出了实现该系统的通讯协议和软件控制流程。其包括终端和服务端之间的通讯，服务器和坐席之间的通讯，可以处理多次重复呼叫，呼叫任务中途更改，短信呼叫变更为语音呼叫，呼叫异常情况。可以对处理情况进行跟踪，处理流程可以自

动核实；能够实现语音报时，终端时间自动核对，信号强弱语音提示，电量语音提示。

1.4 本文工作

由于传统型老年呼叫中心提供的功能，已不能满足用户的需要，只有采用多种通信方式并将其融合到一个平台，提供给用户使用，才能使得老年呼叫中心的服务方式足够灵活、简单，运营成本更低。由于通信是基于手机语音业务，短信(SMS)和手机数据业务(GPRS)，局域网协议，广域网协议实现的，因此本文的一个研究点就是研究如何扩展平台协议栈以支持多种通信方式及其实现其一体化。

SaaS 模式在 CRM、ERP 等企业管理领域得到了很大发展，特别是对于中小企业节约成本来说，SaaS 模式是一个很好的解决方案。SaaS 软件之所以受到业界重视，不仅仅是因为为软件提供形式的转变，最重要的是 SaaS 为业界提供了一整套解决软件生产和消费问题的思想和方法。因此，SaaS 不仅仅是软件行业的一次技术革新，更是催生了一个崭新的商业运营模式。这种创新商业模式的价值和作用无可估量，必将在互联网时代引发一场翻天覆地的软件革命^[1]。本系统涉及到很多从事物流和医疗服务的中小机构，因此本文中在新平台上采用 SaaS 模式，将多种新的技术结合在一起也是一个主要的创新和研究重点。

采用 SaaS 模式，需要将整个平台的展示层以 Web 页面的形式表现出来，使得用户可以通过互联网获得呼叫中心服务。为了得到更好的人机交互性，传统的 web 页面已经无法很好的解决这一问题，就又引出了本文的一个研究重点，在 Web 客户端采用 RIA 的 SilverLight 技术来实现，在不改变原有的 Web 架构的基础上获得类似桌面应用程序的人机交互体验。

综上，引出本文的研究内容：

1. 将 SaaS 模式应用到智能医疗平台中，系统功能以 SilverLight 页面形式提供。可以将音频、视频、文本等多种技术融合起来。
2. 基于 TCP 和 COM 的座席的设计与实现，从而把原来的以语音为中心的呼叫系统无缝连接到智能医疗平台中。
3. 短信(SMS)和手机数据业务(GPRS),GPS 定位和智能医疗平台的自动接入，以及工单处理的自动核对。

1.4 章节安排

第一章：绪论，介绍本课题的背景和现状，作者所做的工作以及论文结构。

第二章：相关技术介绍，通过系统涉及到的几种技术进行介绍和分析，说明如何选择这些技术进行整个系统系统构成。

第三章：系统总体分析和设计，结合 UML、RUP 和设计模式以及传统的软件工程方法，根据需求和技术分析确定设计目标和设计思路，完成总体设计方案，并对系统软构成进行具体说明。

第四章：详细设计及实现，描述了终端与后台软件的设计与实现，对笔者负责的呼叫服务器协议模块进行详细设计、实现。

第五章：系统测试，主要从性能和功能两个方面对前面的设计目标进行验证。

第六章：结束语，总体阐述了本文所做的工作，总结了本系统自身的优缺点及下一步的研究方向。

最后是致谢、参考文献以及本文相关的附录。

第二章 医键通系统相关技术介绍

本章开始将正式进入课题研究,本章的主要内容就是分析研究现有的终端与服务器、电子商务平台的联结方式方面的技术、系统整体架构;为后面的系统设计做好必要的技术积累。主要将明确三个方面的技术分析内容:语音业务、短信(SMS)和手机数据业务(GPRS)接入技术;服务器与电子商务平台连接技术;第三部分重点分析系统架构和组网特点,并确定合适系统架构。智能医疗平台指的是原来完全需要人工操作的工作,有一部分能够由设备自动识别和完成,具有一定的人工智能的特点。本论文课题属于智能医疗平台系统的一部分,其他部分主要由家庭移动监护系统、心脏远程诊断系统、糖尿病与高血压报警与护理系统等组成。目的就是要把现在完全由人工操作,或者需要在现场操作的工作,逐步变成自动操作,可远距离操作和识别的智能医疗系统。

2.1 接入技术

做为系统最重要的技术部分,是需要把原来市场已经存在的传统呼叫中心,和新型的短信呼叫中心,改良的 GPRS 传输结合在一起,形成一个复杂的,建立在统一平台之上的,以短信呼叫为核心的系统;也是本论文分析研究的重点。新的呼叫中心由于采用短信方式或者 GPRS 方式,从而把同步信号转换成异步信号,或者是把模拟语音信号转化成数字信号,从而极大节省了传统 PSTN 所存在的资源瓶颈,在相同的语音带宽的情况下,可以进行数据分拣和分类,通过采用了新技术,通过快速提取并显示用户历史信息,从而帮助坐席工作人员快速处理问题。下面主要对涉及到的这几种技术进行介绍和描述,这几种技术都是常用技术,主要是如何进行统一。

2.1.1 语音业务接入概述

目前的呼叫中心,主要是通过语音通讯模块和管理软件。一般很难判断这种系统使用的开发语言 and 平台,那么如何进行连接呢?一般可以采用两种方式,UDP 和 COM 接口。

UDP 是传统网络编程技术,在此不做过多的讨论;COM 是微软推出的组件标准,它不仅定义了组件程序进行交互的标准,并且提供了组件程序运行所需的环境.COM 对象是二进制可执行代码的共享.面向对象是原代码共享,设计模式是设计的共享。

COM 不仅提供了组件之间的接口标准,还引入面向对象的思想.在 COM 标准中,对象是一个活跃的元素,称为 COM 对象

组件模块为 COM 对象提供了活动的空间,COM 对象以接口的方式提供服务,这种接口称为 COM 接口。其结构如图 2.1。

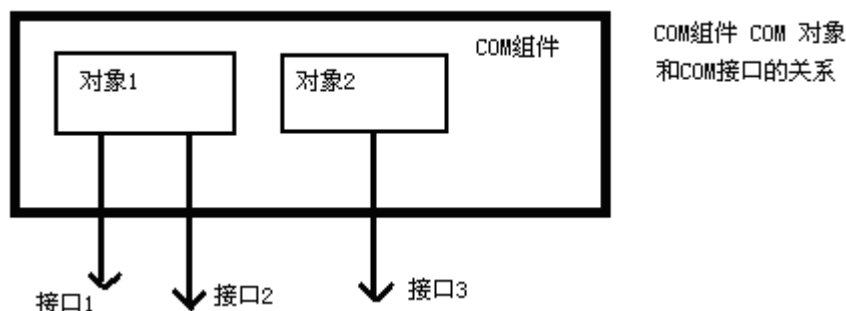


图 2.1 COM 接口示意图

COM 标准包括规范和实现两部分,规范部分定义了组件和组件之间的通信机制,这些机制不依赖任何的特定语言和操作系统,COM 标准的实现部分是 COM 库,COM 库为 COM 规范的具体实现提供了核心服务。

与 C++类似,对象是某个类的实例;类是一组相关的数据和功能结合在一起的一个定义,使用对象的应用称为客户或对象的用户,接口是一组逻辑上相关的函数集合,其函数也称为接口成员函数。按照惯例,接口名常以 “I” 为前缀.对象通过接口成员函数为客户提供各种形式的服务。

在 COM 模型中,对象本身对于客户来说是不可见的,客户请求服务时,只能通过接口进行,每一个接口都由一个 128 位的全局唯一标识符(GUID)来标识,客户通过 GUID 来获取指针,再通过接口指针,客户就可调用其相应的成员函数。所以在 COM 模型中,对象通过接口和接口的函数提供服务。

COM 本身除了规范之外,也有实现的部分,其中包括一些核心的系统级代码,也正是这部分代码,才使对象和客户之间可通过接口在二进制代码级进行交互。包括:

1. 提供少量的 API 函数实现客户和服务端 COM 应用的创建。
2. COM 通过注册表查找本地服务器以及程序名和 CLSID 转换。
3. 提供一种标准的内存控制办法,使应用控制进程中的内存分配。

COM 库一般不在应用程序层实现,而是在操作系统层实现,所以程序员编写代码不需要关心基础代码,调用 API 即可完成。

以下是使用 COM 组件的方法。

```
//创建类工厂  
IClassFactory * pClassFactory=NULL;  
hr=::CoGetClassObject  
( CLSID_MATHCOM,CLSCTX_INPROC,NULL,IID_IClassFactory,  
(void**)&pClassFactory);  
//创建对象  
pClassFactory->CreateInstance  
(NULL,IID_ISimpleMath,(void**)&pSimpleMath);
```

2.1.2 短信服务（SMS）接入概述

短信接入可以采用短信网管和短信猫两种接入方式。

短信网关 ISMG 全称 Internet Short Message Gateway，主要是为了解决各网络、各运营商之间的短信互通和 SP 的接入问题。它为应用单位收发短信而提供的一个动态数据交换平台系统。通过该系统的接口软件，可以将短信平台与各种系统和软件进行无缝高效相连，将应用单位的系统随时产生的动态信息转变成手机短信，通过梦网平台连接移动和联通的短信中心以端口特服号码进行实时中发送和接受，为各种系统（或软件）建立一个快速的短信双向（或单向）通道，以便手机用户采用短信方式与 SP 双向通信，接收 SP 提供的信息服务。

短信网关一般采用 SP 直接与某一个 SMSC (Short Message Service Center 短消息服务中心) 连接，由 SMSC 直接通过信令网发送短信给用户的方式。

短信网关的作用是为 SP 与短消息中心之间数据交换提供一条安全、快捷的通道，以便手机用户利用短信方式与 SP 双向通信，接收 SP 提供的信息服务，同时完成相应计费采集的功能。短信网关包括与负责范围内 SP 进行连接的通讯软件（采用 CMPP 协议）。与负责范围内 SMSC 相连的协议软件（采用 SMPP 协议）。与其他 ISMG 进行连接的通讯软件（采用 CMPP 协议）、业务处理软件、网络管理软件、防火墙软件和为计费提供原始活单的计费处理软件等。

短信网关各组成部分的功能为：

1. SMPP 代理系统遵循 SMPP 3.3 版本协议与 GSM 网中短消息中心连接，实现高效、可靠的数据传输。该系统支持流量控制功能，能够根据 SMSC 的业务量进行发送流量控制。

2. 通信代理系统实现与 SP 等内容供应商的连接和协议互通。它基于 TCP / IP 协议基础之上, 利用 CMPP 协议与 SP 之间建立一条安全、高效的传输通道。该系统支持流量控制功能, 能够根据本身的业务量进行接收流量控制。

3. 防火墙作为短信网关的重要功能组成部分, 其功能是对短信网关内部其它相关模块进行保护, 实现针对内外访问的包过滤和代理。

4. 短消息网关处理系统完成网关的业务处理, 包括: 向汇接网关进行路由查询, 在本地建立短信网关 ID、用户手机号码、SP ID 及其 IP 地址对应表的缓存, 建立用户手机号码段与 SMSC (短信中心) 地址的对应表, 完成对数据分发功能的支持、计费原始话单的提供及处理等。

5. 短信网关计费系统提供短信网关的原始话单记录 (CDR)。

6. 业务管理系统包括业务管理和网管监控功能。

业务管理主要完成对业务的统计报告、生成报表、运营者对用户数据的添加、修改、删除等; 网管监控主要完成对网关系统的监控、查询、操作和维护的管理。

短信网关提供 SNMP 接口、开放接口协议和信息格式、开放 MIB 库, 支持配置、告警、性能数据的采集。业务管理主要完成对业务的统计报告、生成报表、运营者对用户数据的添加、修改、删除等; 短信网关采用防火墙技术, 可以支持 IP 包过滤和应用代理方式, 防止外界的攻击, 实现信息的安全。短信网关与 SP 在进行 CMPP 协议的连接建立时, 采用 MD5 对互相的身份进行认证, 实现业务的安全。

短信猫收发短信的原理, 资费和我们的平常所用的手机是一样的, 但因为短信猫专注于短信收发应用, 所以相对于手机, 短信猫在短信收发的速度要更快, 可靠性更高, 实时发送等优点, 在目前的企业短信中应用广泛。标准短信猫=短信猫硬件+短信猫二次开发包。标准短信猫是短信猫硬件和软件的有机结合体。

短信二次开发包是为了让一些上层语言的开发人员, 或者说对硬件低层的 AT 指令不熟悉的程序员, 快速开发应用程序, 将该短信猫集成到自己的系统中去的一种接口方式。短信二次开发接口有好多种, 比如说 OCX 方式的短信二次开发控件, 还有 DLL 动态链接库的接口方式, 还有封装成中间件或者封装成一个 Server 的形势的数据库接口方式。短信模块, 实现短信收发功能主要电子元件, 是一种基于无线 GSM 技术的工业级的 MODEM 配件, 是一种内嵌 GSM 无线通信模块, 与外围电路及电路板结合, 插入移动运营商的手机 SIM 卡, 可以与移动运营商的短信中心建立无线连接, 本地通过与 PC 的连接可以实现计算机控制应用系统实现自由的短信收发。

基于短信猫的开发应用, 有以下几种方式:

1. 直接使用 AT 指令: 通过串口用 AT 指令驱动短信模块收发短信, 这是最底层的开发模式, 需要对短信模块的 AT 指令相当熟悉;

2. 短信猫开发包：短信猫厂商基于串口 AT 指令集成的二次开发包，开发商只需直接调用短信收发 API 即可；

3. 短信猫通信中间件：短信猫厂商提供的基于数据库接口的短信收发后台服务软件，是一种更高级的短信开发解决方案。

基于短信猫的短信二次开发大都遵循如下开发架构：

1. 短信相关应用需要发送短信时，需要将短信接收者与内容提交到短信发送队列；同时从短信接收队列中读取收到的短信；

2. 软件开发商需要开发独立的短信后台服务，从短信发送队列中读取短信，调用短信猫开发包发送短信；同时通过调用短信猫开发包读取设备已收到的短信，放入短信接收队列；

3. 短信猫开发包内部实际上是通过串口通讯与短信猫连接，通过 AT 指令驱动短信模块收发短信。

因为短信猫是串行通讯设备，必须串行提交短信发送，而且提交后必须等到其有回应后才能提交下一条，否则会造成短信猫死机。特别是现在大部分应用都是多用户应用，如果存在多线程同时并发操作短信模块，也会造成短信猫死机。即使是针对同一短信模块的收发，也必须为一前一后串行，而不能通过收发两个并发线程来操作。因此建议使用短信队列，常用的方式就是使用数据库表。

按照目前方式和系统规模，可在用户量不大的情况下采用短信猫的方式，大约可以一个短信猫解决 10000 人每小时的服务，再以后系统规模扩大后，进行短信网管的接入方案。从设计上可以考虑把系统的接口和实现进行分离，以后出现改变系统就能自动适应。

2.1.3 GPRS 接入概述

GPRS 经常被描述成“2.5G”，也就是说这项技术位于第二代（2G）和第三代（3G）移动通讯技术之间。它通过利用 GSM 网络中未使用的 TDMA 信道，提供中速的数据传递。GPRS 突破了 GSM 网只能提供电路交换的思维方式，只通过增加相应的功能实体和对现有的基站系统进行部分改造来实现分组交换，这种改造的投入相对来说并不大，但得到的用户数据速率却相当可观。而且，因为不再需要现行无线应用所需要的中介转换器，所以连接及传输都会更方便容易。如此，使用者既可联机上网，参加视讯会议等互动传播，而且在同一个视讯网络上（VRN）的使用者，甚至可以无需通过拨号上网，而持续与网络连接。GPRS 分组交换的通信方式在分组交换的通信方式中，数据被分成一定长度的包（分组），每个包的前面有一个分组头（其中的地址标志指明该分组发往何处）。数据传送之前并不需要预先分配信道，建立连接。而是在每一个数据包到达时，根据数据包头中的信息

(如目的地址),临时寻找一个可用的信道资源将该数据报发送出去。在这种传送方式中,数据的发送和接收方同信道之间没有固定的占用关系,信道资源可以看作是由所有的用户共享使用。

由于数据业务在绝大多数情况下都表现出一种突发性的业务特点,对信道带宽的需求变化较大,因此采用分组方式进行数据传送将能够更好地利用信道资源。例如一个进行 WWW 浏览的用户,大部分时间处于浏览状态,而真正用于数据传送的时间只占很小比例。这种情况下若采用固定占用信道的方式,将会造成较大的资源浪费。

在 GPRS 系统中采用的就是分组通信技术,用户在数据通信过程并不固定占用无线信道,因此对信道资源能够更合理地应用。在 GSM 移动通信的发展路标中,GPRS 是移动业务和分组业务相结合的第一步,也是采用 GSM 技术体制的第二代移动通信技术向第三代移动通信技术发展的重要里程碑

目前,用手机上网还显得有些不尽人意。因此,全面的解决方法 GPRS 也就这样应运而生了,这项全新技术可以令您在任何时间、任何地点都能快速方便地实现连接,同时费用又很合理。简单地说:速度上去了,内容丰富了,应用增加了,而费用却更加合理。

1. 高速数据传输

速度 10 倍于 GSM,更可满足您的理想需求,还可以稳定地传送大容量的高质量音频与视频文件,可谓不一般的巨大进步。

2. 永远在线

由于建立新的连接几乎无需任何时间(即无需为每次数据的访问建立呼叫连接),因而您随时都可与网络保持联系,举个例子,若无 GPRS 的支持,当您正在网上漫游,而此时恰有电话接入,大部分情况下您不得不断线后接通来电,通话完毕后重新拨号上网。这对大多数人来说,的确是件非常令人恼火的事。而有了 GPRS,您就能轻而易举地解决这个冲突。

3. 仅按数据流量计费

即根据您传输的数据量(如:网上下载信息时)来计费,而不是按上网时间计费也就是说,只要不进行数据传输,哪怕您一直“在线”,也无需付费。做个“打电话”的比方,在使用 GSM+WAP 手机上网时,就好比电话接通便开始计费;而使用 GPRS+WAP 上网则要合理得多,就像电话接通并不收费,只有对话时才计算费用。总之,它真正体现了少用少付费的原则。

综上所述,GPRS 与短信相比,永远在线决定了数据的实时性,按流量计费决定了成本的有效性,但是其开发技术相对短信要复杂些,需要的网络资源也复杂。

最终,我们选择了以 COM 方式,把传统呼叫中心接入到智能医疗平台,通过 COM 接口可以取到主叫的电话号码,自动生成工单,显示客户基本资料历史情况;

通过短信接入来分类用户请求，并对用户请求生成工单进行处理。最终和系统的电子商务平台联系在一起。

2.2 服务器与电子商务平台连接技术研究

本节从短信、电话拨入、工单处理确认等几个方面对技术进行分析研究。

2.2.1 概述

系统要求当终端短信，或者电话拨入时，能自动识别电话持有的信息，例如 基本信息、姓名、年龄、性别、病史、历史服务、个人脾气和好恶；主要采用的药品和器械，付费状况。

同时要求电话必须录音，坐席能按照一定的规则进行处理用户请求。对超时没有处理的请求能有人进行异常处理。

系统能够自动对用户的请求是否得到坐席人员和社区服务人员的处理进行相关统计和处理。

2.2.2 系统逻辑划分

系统主要由终端、呼叫中心、业务处理 3 部分组成。其中呼叫中心分为电话和短信部分；业务处理分为 UI 层、业务层和支持层（引擎与适配）；同时系统支持 WEB 与 TCP/IP 协议，具体如图 2.2 所示。

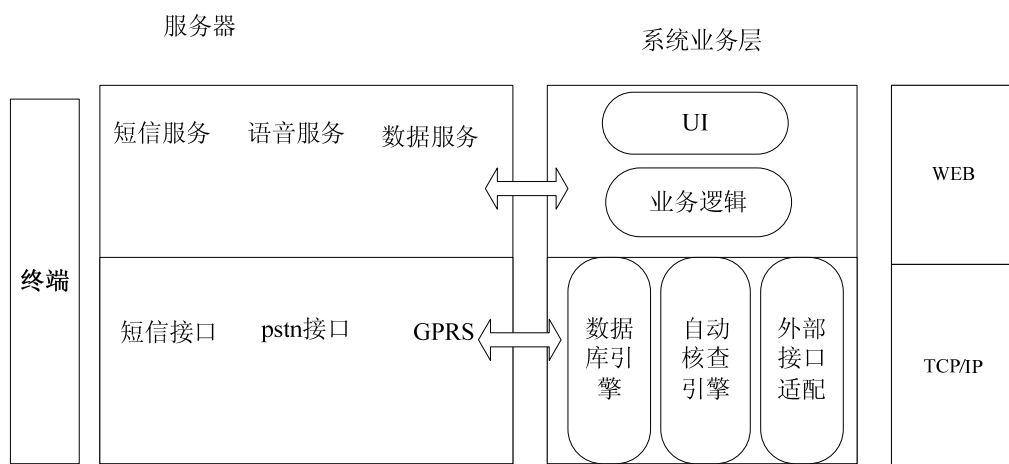


图 2.2 系统逻辑层次示意图

1. 终端可以通过短信网管和 pstn 接口完成与服务器的数据交互。
2. 服务器通过短信服务和语音服务模块完成相关数据和业务系统的接口。
3. 业务部分由三部分组成，采用两种协议。

4. 为了在短信处理和 PSTN 接口处业务逻辑保持一致, 采用适配方法完成。

5. 由于系统需要在呼叫中心内部以及各个社区之间信息交互, 所以采用 web 协议和 TCP/IP 协议。

2.2.3 系统协议说明

PSTN:

所谓公用电话交换网 (PSTN——Public Switch Telephone Network), 即我们日常生活中常用的电话 PSTN 网。众所周知, PSTN 是一种以模拟技术为基础的电路交换网络。在众多的广域网互连技术中, 通过 PSTN 进行互连所要求的通信费用最低, 但其数据传输质量及传输速度也最差, 同时 PSTN 的网络资源利用率也比较低。通过 PSTN 可以实现的访问:

拨号上 Internet/Intranet/LAN; 两个或多个 LAN 之间的网络互连;

和其它广域网技术的互连 尽管 PSTN 在进行数据传输时存在这样或那样的问题, 但这是一种仍不可替代的联网介质(技术)。特别是 Bellcore 发明的建立在 PSTN 基础之上的 xDSL 技术和产品的应用拓展了 PSTN 的发展和应用空间, 使得联网速度可达到 9Mbps~52Mbps 之间。

PSTN 入网方式

PSTN 的入网方式比较简便灵活, 通常有以下几种:

通过普通拨号电话线入网。

只要在通信双方原有的电话线上并接 Modem, 再将 Modem 与相应的上网设备 PSTN 备相连即可。目前, 大多数上网设备, 如 PC 或者路由器, 均提供有若干个串行端口, 串行口和 Modem 之间采用 RS-232 等串行接口规范。这种连接方式的费用比较经济, 收费价格与普通电话的收费相同, 可适用于通信不太频繁的场合。

通过租用电话专线入网。

与普通拨号电话线方式相比, 租用电话专线可以提供更高的通信速率和数据传输质量, 但相应的费用也较前一种方式高。使用专线的接入方式与使用普通拨号线的接入方式没有太大的区别, 但是省去了拨号连接的过程。通常, 当决定使用专线方式时, 用户必须向所在地的电信局提出申请, 由电信局负责架设和开通。

经普通拨号或租用专用电话线方式

由 PSTN 转接入公共数据交换网 (X.25 或 Frame-Relay 等) 的入网方式。利用该方式实现与远地的连接是一种较好的远程方式, 因为公共数据交换网为用户提供可靠的面向连接的虚电路服务, 其可靠性与传输速率都比 PSTN 强得多。

短信

短信 short message service, 简称 SMS 是用户通过手机或其他电信终端直接发

送或接收的文字或数字信息，用户每次能接收和发送短信的字符数，是 160 个英文或数字字符，或者 70 个中文字符。

短信收发过程

短信是伴随数字移动通信系统而产生的一种电信业务，通过移动通信系统的信令信道和信令网，传送 手机短信

文字或数字短信息，属于一种非实时的、非语音的数据通信业务。短信可以由移动通信终端(手机)始发，也可由移动网络运营商的短信平台服务器始发，还可由与移动运营商短信平台互联的网络业务提供商 SP(包括 ICP、ISP 等)始发。

从实现短信业务功能的技术手段而言，通过手机终端发送和接收点对点消息虽然占据主流地位，但并非唯一形式，固定电话、小灵通以及互联网正在成为新的工具和载体。

在短信业务发展之初，短信的发送和接收是手机开始，最终到手机，中间经过 GSM 网/CDMA 网、短信中心等网络要素;随着短信点播及定制业务的出现，手机和互联网站、短信增值业务系统可以互相发送短信，互联网站和短信增值业务系统成为新的网络要素，GSM/CDMA 网、短信中心仍保持着短信网络核心地位;随着以固定电话网为基础的“小灵通”手机在中国出现，短信业务扩展到固定网，终端不再局限于手机，小灵通手机或特殊固定终端都可以成为短信的发送和接收方，网络基础也不再局限于 GSM/CDMA 网等数字移动网，固定网也可以成为短信的网络基础，短信网络要素发生了显著变化。

伴随着短信从手机扩展到小灵通及固定终端、从数字移动通信网扩展到固定电话网，人们对短信的认识也不再仅看作是数字手机的“专利”，业务形态在改变、网络要素在变化、信息内容在丰富，这一过程中始终不变的只有两点：一是短信的信息长度，始终是不超过 160 个英文或数字字符，或 70 个汉字，这与短信基于通信系统的信令网传送内容的机制密切相关。二是短信传递的方式——存储转发，当用户无法接收时，短信不会丢失，暂时存放在短信中心，当用户重新登录进网的时候，短信会迅速递交到用户手机上。

这些与生俱来的特点，使短信具备了传递准确可靠、迅速及时的优点，使短信具备了影响人们的习惯的基本条件! 短信作为手机上的一种业务，让本来具有语言传递功能的手机变成了电报式的解读工具，让耳朵闲置，让文字彰显更大的作用;短信作为用文字传递信息和沟通的一种方式，让拇指灵敏发达起来，让嘴巴休息，现代电信的业务具有了技术和文化的双重色彩。

2.2.4 RIA 和 Silverlight 技术介绍

为了在社区或者其他服务商的工作环境提供更加流畅和高体验的 web 效果，

系统采用 Silverlight 这种 RIA 技术做为支撑。

传统网络程序的开发是基于页面的、服务器端数据传递的模式，把网络程序的表示层建立于 HTML 页面之上，而 HTML 是适合于文本的，传统的基于页面的系统已经渐渐不能满足网络浏览者的更高的、全方位的体验要求了，这就是被 Macromedia 公司称之为的“体验问题”("Experience Matters")，而富因特网应用程序(Rich Internet Applications, 缩写为 RIA)的出现也就是为了解决这个问题。

RIA 技术只是一类技术的统称，有很多种不同的实现方案，比较常见的包括 Adobe 公司的 Flex、Sun 公司的 JavaFX、微软公司的 Avalon 和 SilverLight 等。

微软 Silverlight 是一个跨浏览器、跨客户平台的技术，能够设计、开发和发布有多媒体体验与富交互(RIA,Rich Interface Application)的网络交互程序。因为 Silverlight 提供了一个强大的平台，能够开发出具有专业图形、音频和视频的 Web 应用程序，增强了用户体验，所以 SilverLight 吸引了设计人员和开发人员的眼球。同时，Silverlight 还提供了强大的工具来提高他们的工作效率^[2]。

Silverlight 将多种技术结合到一个开发平台，可以在其中选择符合需求的合适的工具和编程语言。SilverLight 提供了如下的特性：

1. WPF 和 XAML。Silverlight 包含了 Windows Presentation Foundation(WPF)技术，这个技术在创建用户界面时极大的扩展了浏览器元素。WPF 可以创建融合图形、动画、媒体和其他的富客户端特性，扩展了基于浏览器的用户界面，超越了 HTML 所提供的。可扩展应用程序标记语言(XAML)提供了创建 WPF 元素的声明性标记。

2. 对于 JavaScript 的扩展。Silverlight 提供了对于全球浏览器脚本语言的扩展，从而为浏览器用户界面提供更加强大的控件，包括与 WPF 元素工作的能力。

3. 跨浏览器，跨平台的支持。Silverlight 在所有的流行浏览器(任何平台)运行一致。设计和开发应用程序不需要担心你的用户是什么平台什么浏览器。

4. 与现存应用程序的集成。Silverlight 可以与已经存在的 JavaScript 和 ASP.NET AJAX 代码无缝集成，不会使已经创建的功能缺失。

5. 可访问.NET Framework 编程模型和相关工具。可以使用托管的 Jscript 和 IronPython 或者 C#和 VB 这样的动态语言来创建基于 SilverLight 的应用程序。可以使用 Visual Studio 这样的开发工具来创建基于 Silverlight 的应用程序。

6. LINQ。Silverlight 包含集成查询(LINQ)语言。

7. 如果已经使用 ASP.NET，可以将 Silverlight 集成到熟悉的 ASP.NET 服务器和客户端功能。可以在 ASP.NET 中创建基于服务器的资源，使用 ASP.NET 的 AJAX 特性与服务器端资源交互而不会打断用户。

Silverlight 已经是使用中的一个广泛平台，可同时针对应用程序和单纯的媒体案例 (包括了 HD 质量、透过 Smooth Streaming 的互动视讯) 建立丰富的经验。

Silverlight 4 提供额外的功能来让您建立更丰富、更吸引人的高效能互动经验和创新的媒体经验：

流畅的接口增强可透过动画效果提升应用程序的使用性；网络摄影机和麦克风可让您在聊天或客户服务应用程序等情况中分享视讯和音频；音频和视讯本机录制功能无须服务器互动即可撷取 RAW 视讯，因而启用了众多的用户互动和通讯案例，例如视频会议；以复制和贴上或拖放等功能将数据放到您的应用程序内。长列表现在可以用鼠标滚轮轻松地卷动；透过右键内容菜单等新功能支持传统的桌面互动模型；支持 Google 的 Chrome 浏览器；效能优化意指 Silverlight 4 应用程序可以更快地启动，并且执行速度是相等的 Silverlight 3 应用程序的 200%。多重触控支持可将许多笔势和触控互动整合至使用者经验之中；多播网络可让企业降低串流广播事件的成本，例如公司会议和训练，并且完美地与现有的 Windows Media Server 串流基础结构互通；PlayReady 增强了以 Silverlight DRM 来保护 H.264 媒体的内容；音频/视讯串流的输出保护可让内容拥有者或散发者确保受保护的内容只能经由安全的视讯联机来检视。

2.3 系统整体架构技术研究

本节从整体概述，组网方式，数据库，短信收发，终端产品,整体的网络规划几个方面进行分析和研究。

2.3.1 组网方式

从以下的应用场景，图 2.3 和图 2.4 中可以知道，需要在呼叫中心设立一个坐席中心，中心服务器和坐席之间，可以通过 TCP/IP 协议，在局域网内完成数据库访问，消息通讯，坐席调度等操作。

通过以上结构图的研究，我们发现首先系统存在局域网和互连网两种网络结构，其次，对于多个社区服务中心，就存在分布式情况，而在互连网上，真实 IP 的资源是昂贵的，同时给程序设计又带来很大的问题，增加了程序设计的复杂度和工作量，若不采用真实 IP,可以通过域名解析的方式实现，一个公司，申请众多的域名，也是不合适的，如何解决这个问题，也是一个难点。

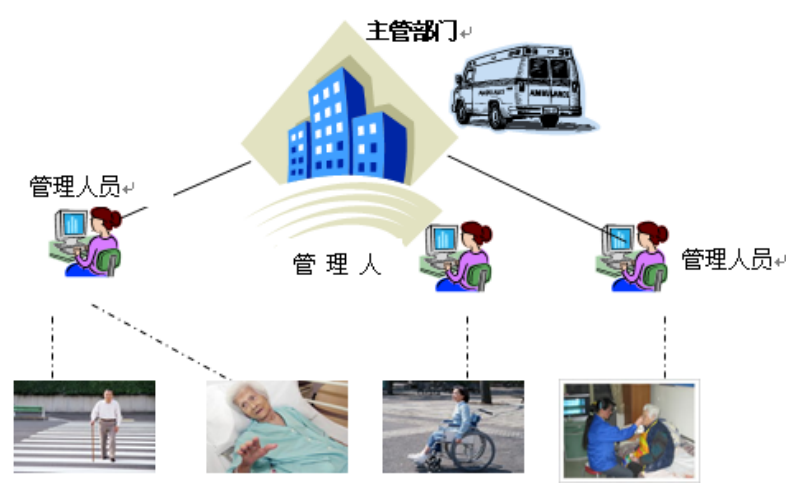


图 2.3 老年服务中心或老干部活动中心

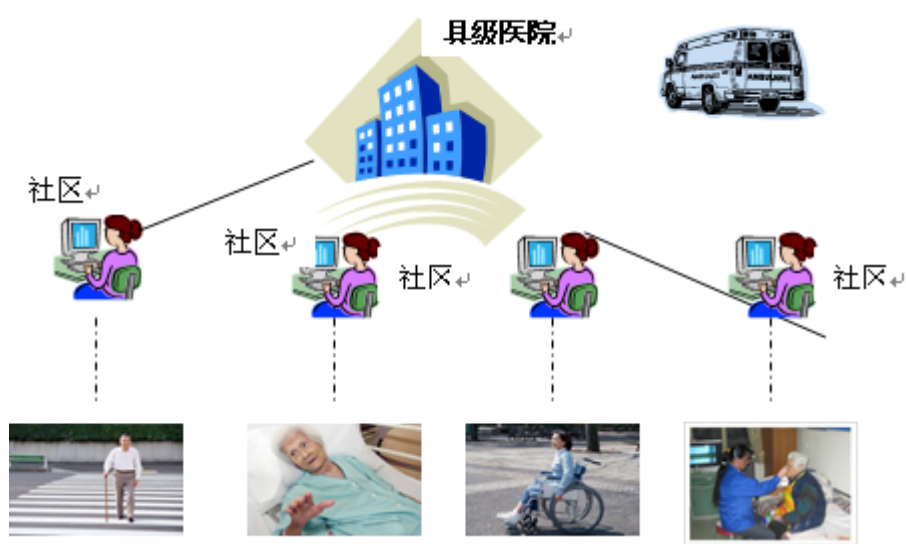


图 2.4 以县为单位的管理结构

对于以上的问题，就是 SaaS 中经常遇到的问题，如何把成本和技术有效的结合，成为本系统的一个难点。

对于这个问题，我们采用呼叫中心外包的方式，具体的讲。就是寻找一家呼叫中心公司，把我们的系统放在其内部，由于呼叫中心存在服务器托管中心，所以我们采用 IP+端口的方式，来解决这个问题，同时再把语音服务通过 IP 电话+400 的方式，接入到我们的系统。从而解决以上问题。

2.3.2 设备终端

手表式终端，便于携带。具有通话、短信、电量提示、报时、信号提示。早

期采用购买通信芯片和单片机控制，主要技术难点为 AT 指令的掌握。经过测试发现问题：体积大，质量不可靠。10 台设备样机只有 4 台可以使用，最终只有两台可以使用。而且整机价格超过 150 元。采用 MTK 方案就不存在这些问题，价格只有原来的 2/3。

MTK 平台是联发科公司推出的一种新型的手机设计解决方案，到目前为止，它已经成为了国产手机生成厂商首选的平台，其高度的模块集成特性、快捷的出货速度和低廉的价格优势催生了中国大陆数量庞大的手机研发公司，它创造了这个时代最响亮的名词：“山寨”，并引领了从 2006 年以来山寨机的潮流^[3]。

2.3.3 短信接入

可以采用短信猫和短信网管两种接入方式。这主要涉及到短信协议和网管接入协议，按照目前方式和系统规模，可在用户量不大的情况下采用短信猫的方式，大约可以一个短信猫解决 10000 人每小时的服务，在以后系统规模扩大后，进行短信网管的接入方案。从设计上可以考虑把系统的接口和实现进行分离，以后出现改变系统就能自动适应。

2.3.4 数据库部分

数据库本身压力不大，目前 Oracle ,SQL Server,DB2,MySql 都可以使用。由于服务器采用 windows server2003 或者以上版本。所以主要考察 SQL Server 和 MySql 两种，由于 MySQL 被 Oracle 收购，目前种种迹象表明 MySql 将不是开源的首选，以后维护和升级存在教多问题，所以决定采用 SQL Server。

2.3.5 呼叫中心内部结构图网络规划介绍

整个系统包括以下内容：

1. 前台呼叫终端。
2. 呼叫中心服务器与数据库。
3. 坐席处理；包括电话和短信。
4. 呼叫中心的硬件与软件（排队处理，通话录音等）。
5. 接受任务的团体或个人终端电脑（要求联网）。
6. 呼叫延迟或异常处理处理平台。
7. 短信设备或直接与运营商服务器连接。

(单口短信猫:1900 元/套 ,发送速度:750 条/小时,接受 1 小时 10000 条)。最终网

络规划图如图 2.5 所示。

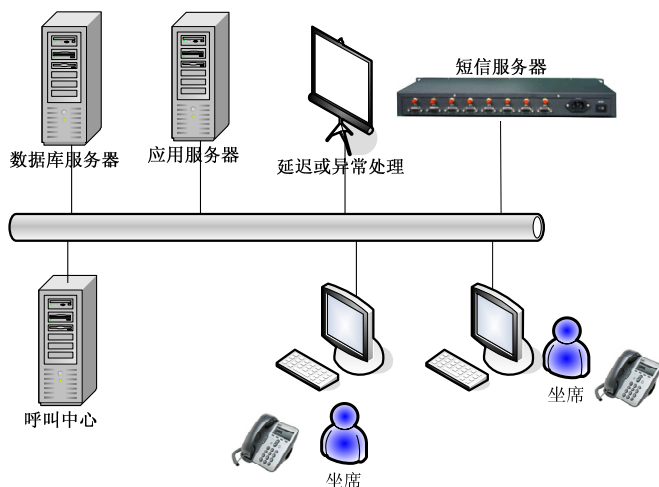


图 2.5 内部网络规划

2.3.6 UML 概述

UML 是一种定义良好、易于表达、功能强大且普遍适用的建模语言。它溶入了软件工程领域的新思想、新方法和新技术。它的作用域不限于支持面向对象的分析与设计，还支持从需求分析开始的软件开发的全过程。面向对象技术和 UML 的发展过程可用图形来表示，标准建模语言的出现是其重要成果。在美国，截止 1996 年 10 月，UML 获得了工业界、科技界和应用界的广泛支持，已有 700 多个公司表示支持采用 UML 作为建模语言。1996 年底，UML 已稳占面向对象技术市场的 85%，成为可视化建模语言事实上的工业标准。1997 年 11 月 17 日，OMG 采纳 UML 1.1 作为基于面向对象技术的标准建模语言。UML 代表了面向对象方法的软件开发技术的发展方向，具有巨大的市场前景，也具有重大的经济价值和国防价值。UML 是一个标准的图形表示法，它不是面向对象的分析和设计，也不是一种方法，它仅仅是一组符号而已^[4]。

标准建模语言 UML 的重要内容可以由下列五类图（共 9 种图形）来定义：第一类是用例图，从用户角度描述系统功能，并指出各功能的操作者。

第二类是静态图 (Static diagram)，包括类图、对象图和包图。其中类图描述系统中类的静态结构。不仅定义系统中的类，表示类之间的联系如关联、依赖、聚合等，也包括类的内部结构（类的属性和操作）。类图描述的是一种静态关系，在系统的整个生命周期都是有效的^[5]。

对象图是类图的实例，几乎使用与类图完全相同的标识。他们的不同点在于对象图显示类的多个对象实例，而不是实际的类。一个对象图是类图的一个实例。

由于对象存在生命周期，因此对象图只能在系统某一段时间段存在。包由包或类组成，表示包与包之间的关系。包图用于描述系统的分层结构^[6]。

第三类是行为图（Behavior diagram），描述系统的动态模型和组成对象间的交互关系。其中状态图描述类的对象所有可能的状态以及事件发生时状态的转移条件。通常，状态图是对类图的补充。在实用上并不需要为所有的类画状态图，仅为那些有多个状态其行为受外界环境的影响并且发生改变的类画状态图。而活动图描述满足用例要求所要进行的活动以及活动间的约束关系，有利于识别并行活动^[7]。

第四类是交互图（Interactive diagram），描述对象间的交互关系。其中顺序图显示对象之间的动态合作关系，它强调对象之间消息发送的顺序，同时显示对象之间的交互；合作图描述对象间的协作关系，合作图跟顺序图相似，显示对象间的动态合作关系。除显示信息交换外，合作图还显示对象以及它们之间的关系。如果强调时间和顺序，则使用顺序图；如果强调上下级关系，则选择合作图。这两种图合称为交互图^[8]。

第五类是实现图（Implementation diagram）。其中构件图描述代码部件的物理结构及各部件之间的依赖关系。一个部件可能是一个资源代码部件、一个二进制部件或一个可执行部件。它包含逻辑类或实现类的有关信息。部件图有助于分析和理解部件之间的相互影响程度。配置图定义系统中软硬件的物理体系结构。它可以显示实际的计算机和设备（用节点表示）以及它们之间的连接关系，也可显示连接的类型及部件之间的依赖性。在节点内部，放置可执行部件和对象以显示节点跟可执行软件单元的对应关系^[9]。

从应用的角度看，当采用面向对象技术设计系统时，首先是描述需求；其次根据需求建立系统的静态模型，以构造系统的结构；第三步是描述系统的行为。其中在第一步与第二步中所建立的模型都是静态的，包括用例图、类图（包含包）、对象图、组件图和配置图等五个图形，是标准建模语言 UML 的静态建模机制。其中第三步中所建立的模型或者可以执行，或者表示执行时的时序状态或交互关系。它包括状态图、活动图、顺序图和合作图等四个图形，是标准建模语言 UML 的动态建模机制。因此，标准建模语言 UML 的主要内容也可以归纳为静态建模机制和动态建模机制两大类^[10]。

2.3.7 RUP 概述

首先，RUP 是一套面向对象方法学。RUP 是以面向对象方法为基础的方法学，在业务建模、需求、分析设计、实现、测试等各个规程中，面向对象方法贯穿始

终。即便是在与系统构建关系不大的业务工程中，RUP 也使用 OOA/D 人员熟悉的面对象方法。因此，RUP 更适合 OO 类项目。

RUP 坚持以用例驱动，以架构为中心，迭代和增量的开发方法。用例驱动既做到了以客户为中心，以客户的角度看系统，为客户创建真正可用的系统的构造方式，又解决了传统面向对象方法面临的无法有效的从需求中提取对象的困难，从而填平了需求分析方法和对象设计方法之间的鸿沟。以架构为中心，坚持综合考虑软件系统的各个方面，并优先解决各个方面的主要问题，同时创建常见问题的通用解决方案，从而为解决软件项目的主要风险、准确估算项目进度、提高软件复用、保证软件的整体风格打下了坚实基础。迭代和增量式开发，以风险为驱动，分阶段针对不同的风险制定对策，以保证软件项目的成功完成^[11]。

除此之外，RUP 还吸收了各家所长，以契约设计、测试驱动方法保证软件质量，以架构模式、设计模式增强系统的可扩展性和易维护性，以统一变更管理加强代码等资源质量，以组件架构提高软件复用等。

其次，RUP 是一种适用范围较广的适应性软件过程。

它定义了进行软件开发的工作步骤，亦即定义了软件开发过程中的，什么时候做，做什么，怎么做，谁来做的问题，以保证软件项目有序地、可控地、高质量地完成。

RUP 是一种适应性软件过程，区别于瀑布模型类的预见性软件过程。RUP 不假设从一开始就可以掌握软件开发的全过程，而是坚持以迭代方式推进软件开发，结合不断演进的项目状态和现实变化做出相应的调整，制定出新的计划。实践证明适应性过程比预见性过程更能保证项目成功^[12]。

另一方面，RUP 并没有对软件开发的规范化程度做出明确规定。换言之，RUP 允许我们根据项目的实际情况，对其进行裁剪，以决定哪些文档、过程是必须的。从而为我们在敏捷式开发和规范化开发的选择上提供了灵活性。这一点与以 XP 为代表的敏捷过程有所不同，后者除了强调适应性开发外，也强调尽可能减少文档数量和不必要的管理，以提高主要活动（即软件开发活动）的效率。目前，敏捷过程在产品周期短、变化快和规模相对小的项目中倍受推崇。对 RUP 做适度裁剪，轻装上阵，亦可成为敏捷过程。同理，对 RUP 进行调整也可以使其符合 CMM 或 CMMI 的要求，配置成规范化的过程，从而应对规模较大、相对复杂、但变化不快的软件项目^[13]。

最后，RUP 本身也是一个提供了可定制框架的软件过程产品。

RUP 提供了一个可定制软件产品，其中包括 RUP 方法学指导、过程定义和文档模板，也还包括一些示例工程。除此之外，为了支持 RUP 的方法学理论和最佳实践，Rational 还提供了一系列 CASE 产品，如 RequisitePro(需求管理)、ROSE(可视化建模工具)、ClearCase&ClearQuest(配置和变更管理工具)等。

RUP 四个阶段

1. 初始阶段

初始阶段有时也称先启阶段。初始阶段的目标是为系统建立商业案例并确定项目的边界。为了达到该目的必须识别所有与系统交互的外部实体，在较高层次上定义交互的特性。本阶段具有非常重要的意义，在这个阶段中所关注的是整个项目进行中的业务和需求方面的主要风险。对于建立在原有系统基础上的开发项目来讲，初始阶段可能很短。

初始阶段结束时是第一个重要的里程碑：生命周期目标(Lifecycle Objective)里程碑。生命周期目标里程碑评价项目基本的生存能力。

2. 细化阶段

细化阶段的目标是分析问题领域，建立健全的体系结构基础，编制项目计划，淘汰项目中最高风险的元素。为了达到该目的，必须在理解整个系统的基础上，对体系结构作出决策，包括其范围、主要功能和诸如性能等非功能需求。同时为项目建立支持环境，包括创建开发案例，创建模板、准则并准备工具。

细化阶段结束时第二个重要的里程碑：生命周期结构(Lifecycle Architecture)里程碑。生命周期结构里程碑为系统的结构建立了管理基准并使项目小组能够在构建阶段中进行衡量。此刻，要检验详细的系统目标和范围、结构的选择以及主要风险的解决方案。

3. 构造阶段

在构建阶段，所有剩余的构件和应用程序功能被开发并集成为产品，所有的功能被详细测试。从某种意义上说，构建阶段是一个制造过程，其重点放在管理资源及控制运作以优化成本、进度和质量。

构建阶段结束时是第三个重要的里程碑：初始功能(Initial Operational)里程碑。初始功能里程碑决定了产品是否可以在测试环境中进行部署。此刻，要确定软件、环境、用户是否可以开始系统的运作。此时的产品版本也常被称为“beta”版。

4. 交付阶段

交付阶段的重点是确保软件对最终用户是可用的。交付阶段可以跨越几次迭代，包括为发布做准备的产品测试，基于用户反馈的少量的调整。在生命周期的这一点上，用户反馈应主要集中在产品调整，设置、安装和可用性问题，所有主要的结构问题应该已经在项目生命周期的早期阶段解决了。

在交付阶段的终点是第四个里程碑：产品发布(Product Release)里程碑。此时，要确定目标是否实现，是否应该开始另一个开发周期。在一些情况下这个里程碑可能与下一个周期的初始阶段的结束重合。

2.3.8 设计模式

设计模式是一套理论，由软件界的先辈们（The Gang of Four：包括 Erich Gamma、Richard Helm、Ralph Johnson、John Vlissides）总结出的一套可以反复使用的经验，它可以提高代码的可重用性，增强系统的可维护性，以及解决一系列的复杂问题。做软件的人都知道需求是最难把握的，我们可以分析现有的需求，预测可能发生的变更，但是我们不能控制需求的变更。问题来了，既然需求的变更是不可控的，那如何拥抱变化呢？幸运的是，设计模式给了我们指导，专家们首先提出了 6 大设计原则，但这 6 大设计原则仅仅是一系列“口号”，真正付诸实施还需要有详尽的指导方法，于是 23 种设计模式出现了^[13]。

在详细设计的过程中，我们在一些主要的地方使用了设计模式来增强程序的灵活性和可维护性，以及一些特殊要求。

2.3.9 SaaS

SaaS 是一种软件布局模型，其应用专为网络交付而设计，便于用户通过互联网托管、部署及接入。SaaS 应用软件的价格通常为“全包”费用，囊括了通常的应用软件许可证费、软件维护费以及技术支持费，将其统一为每个用户的月度租用费。对于广大中小型企业来说，SaaS 是采用先进技术实施信息化的最好途径。但 SaaS 绝不仅仅适用于中小型企业，所有规模的企业都可以从 SaaS 中获利。

SaaS 服务模式与传统许可模式软件有很大的不同，它是未来管理软件的发展趋势。相比较传统服务方式而言 SaaS 具有很多独特的特征：SaaS 不仅减少了或取消了传统的软件授权费用，而且厂商将应用软件部署在统一的服务器上，免除了最终用户的服务器硬件、网络安全设备和软件升级维护的支出，客户不需要除了个人电脑和互联网连接之外的其它 IT 投资就可以通过互联网获得所需要软件和服务。此外，大量的新技术，如 Web Service，提供了更简单、更灵活、更实用的 SaaS。另外，SaaS 供应商通常是按照客户所租用的软件模块来进行收费的，因此用户可以根据需求按需订购软件应用服务，而且 SaaS 的供应商会负责系统的部署、升级和维护。而传统管理软件通常是买家需要一次支付一笔可观的费用才能正式启动。

SaaS 企业管理软件分成两大阵营：平台型 SaaS 和傻瓜式 SaaS。平台型 SaaS 是把传统企业管理软件的强大功能通过 SaaS 模式交付给客户，有强大的自定制功能。傻瓜式 SaaS 提供固定功能和模块，简单易懂但不能灵活定制的在线应用，用户也是按月付费。

一般而言，平台型 SaaS 更适合企业的发展，因为它强大的自定制功能能满足企业的应用，当然，并非所有 SaaS 厂商的产品都具有自定制功能，所以企业在选择产品时要先考察清楚。目前业内平台型做的较好的厂商有八百客、Salesfoece 等，

自定制平台，无需编写代码、无需数据库知识、只要深刻理解企业业务，就能实现任何所需，且无需自行维护。其人性化的地方体现在：不同阶段会给企业提供相应的免费试用优惠，让企业真正做到“先使用、后付款”，避免了盲目购买。傻瓜式 SaaS 的功能是固定的，在某个阶段能适应企业的发展，一旦企业有了新的发展，它的无法升级和无自定制的缺点就会暴露出来，这时企业只能进行“二次购买”。平台型 SaaS 和傻瓜式 SaaS 的共同点是都能租赁使用。但是无论是平台型 SaaS 或傻瓜式 SaaS，SaaS 服务提供商都必须有自己的知识产权，所以企业在选择 SaaS 产品时应当了解服务商是否有自己的知识产权。

2.4 本章小结

本章主要对整个系统所涉及的技术进行了分析总结，通过对短信，语音接入，网络通讯，数据库访问等问题进行分析，来完成组网和协议架构，满足需求配置，同时对后面章节涉及到的 UML 和 RUP，设计模式等内容做了简单的概述。

第三章 系统总体分析与设计

本章在上面章节介绍的技术的基础上，分析了基于智能医疗平台的医键通系统所要完成的主要工作。首先，对整个平台系统的功能和性能进行了分析，并在此基础上设计了终端设备的设计要求和平台的系统架构及其网络结构。接着，按照功能需求对系统进行了模块划分，描述了系统中各模块的功能和作用。为系统实现中的详细设计、编码以及提供了基础。本章首先阐述了系统设计目标和要求内容，然后根据要求从系统设计思路、架构组网、功能模块、数据库和性能要求等方面进行详细说明。

3.1 系统总体设计

智能医疗平台是一个终端和后台软件相结合的系统，其主要目的就是通过特制的终端，以及语音呼叫系统的 COM 接口、短信收发、局域网协议、Silverlight 等技术，同时借助一套设计好的自动工单核定流程，让用户的请求一旦发出，就能够让系统自动在业务流程的框架中得到工作状态的核定，而不会由于人的原因造成请求没有及时处理。同时对于没有及时处理的请求，系统能够提供一定的警告和统计能力，帮助系统在运营过程中可以进行人工调整。

系统设计目标如下：

1. 用户佩带腕式终端，所有请求和业务，用户使用腕式终端即可完成。终端参数由短信设置完成。终端要求携带方便，电池可保持 1 周的续航能力。
2. 当用户在遇到紧急情况下，按下“语音紧急呼叫”按钮，终端设备自动拨通以前初始化在终端中的传统语音呼叫中心的电话号码，呼叫中心按照原来呼叫中心的算法完成调度，把相关信息发送到坐席的 COM 控件中，COM 控件自动把消息发送到坐席窗口，坐席窗口调用外部接口适配模块，生成语音请求信息，再把相关信息发送给业务逻辑模块，业务逻辑模块调用数据库引擎得到用户的基本信息和历史信息，显示给坐席人员，同时生成一条请求核对短信，进入业务处理过程。
3. 当用户在遇到紧急情况下，按下“短信紧急呼叫”按钮，终端设备自动发送请求短信到以前初始化在终端中的电话号码，短信服务器调用短信接口得到短信信息，执行短信的算法完成调度，把相关信息发送到坐席的窗口，坐席窗口再把相关信息发送给业务逻辑模块，业务逻辑模块调用数据库引擎得到用户的基本信息和历史信息，显示给坐席人员，进入业务处理过程。
4. 当用户在遇到需要帮助情况下，按下“日常短信请求”按钮，终端设备自动发送请求短信到以前初始化在终端中的电话号码，短信服务器调用短信接口得

到短信信息，执行短信的算法完成调度，把相关信息发送到坐席的窗口，坐席窗口再把相关信息发送给业务逻辑模块，业务逻辑模块调用数据库引擎得到用户的基本信息和历史信息，显示给坐席人员，进入业务处理过程。

5. 业务处理过程：

当用户请求显示出来后，坐席人员需要拨打用户电话，询问用户具体的服务内容（用户直接拨通语音电话不用），当询问内容完成后，坐席人员挂掉电话，终端会自动发送一条请求确认短信（用户没有在规定时间内询问用户，该工单就会产生一条异常记录，自动发送到监控终端，由监控人员进行处理），该短信经过短信服务器，发送到正在处理的坐席的窗口，窗口调用自动核查引擎，改变请求工单的请求状态为派遣状态。这时 UI 部分自动跳转到工单派遣界面，用户选择相关小区的服务商提供服务。然后把工单派遣到相关处理单位的计算机上。

具体处理的服务商的终端会自动显示相关的工单，服务商要再次询问用户，通话结束后，终端会自动发送一条确认信息到短信服务器，短信服务器再发送到对应坐席窗口，窗口调用自动核查引擎，工单自动进入服务状态。用户处理完毕后填写处理报告（若没有及时处理，坐席会产生异常记录，坐席人员会打电话给服务商来解决问题）。

当坐席人员看到服务商服务报告填写完成后，会打电话回访用户，电话结束后会产生回访确认消息，短信服务器会发该消息发送给对应坐席窗口，窗口再次调用自动核查引擎，把工单归档，这时工单处理彻底完成。坐席人员给服务商打分。进入以后的统计数据中。

6. 系统需要提供异常警告机制，保证异常数据能够被处理，警告要以声，光，颜色，动画等多种方式进行提示。

7. 系统提供统计功能，能够对所有异常进行分析，从而管理服务商，坐席，专席，专家等对用户的服务。

8. 系统需要提供系统管理，数据备份，参数管理等功能。

9. 系统需要提供收费管理和用户管理等功能。

3.2 终端的设计主要内容

终端采用 MTK 平台，由于 MTK 平台是高度集成的，所以主要设计在软件和外观设计，其中外观设计主要有工业设计公司完成，电路和软件也由专业设计公司来完成，所以本部分的设计主要表现为对终端传输协议的设计，功能需求设计，以及外观设计。

外观设计要求：

1. 要求为腕式手表外观，体积要小巧，便于携带，体积以 MTK 硬件为参考。

2. 要求模具外壳有一定的抗震, 抗摔, 防水能力。
3. 电池部分要拆卸方便, 满足硬件电路功耗对电池的的要求的基础上, 尽量控制电池的体积。
4. 要求外观设计为原创, 不得抄袭, 以便完成以后的相关专利的申请。
5. 要求各个部件在以后装配时达到最容易的效果, 从而为批量生产创造好的条件。
6. 配件选用目前流行的材料, 同时关注批量时生产成本。
7. 在厚度无法改变的情况下, 产品外观要求从视觉上足够的薄, 借鉴相关视觉设计经验。
8. 按键一定要灵活, 同时要从模具保证具有防意外碰触的能力。

终端设计要求

通讯方式:

数据信息以 SMS 文本方式通过点对点或 GPRS 方式上传或下传, 使用那种传送方式, 可通过设置控制, 同时具备一种方式即可。

功能说明:

1. 开机: 按下开关机键, 开启终端 (手机), 按照预定设置初始化相关的信息。
2. 关机: 长按开关机键, 关闭终端。
3. 呼入: 被叫呼入时, 根据设置的允许号码信息查找, 是允许号码则报来电号码, 报完号码后响铃提示, 不是允许号码直接拒接。如无设置允许号码数据信息则视为允许所有号码呼入, 直接报来电号码后响铃提示, 无来电号码直接响铃。
4. 呼出: 主叫呼出, 本设备只允许呼叫已设置的固定号码, 不允许任意呼出号码, 当按下相应的按键后则呼叫该号码。
5. 应答: 被叫呼入, 有呼入提示后按紧急求助键应答接听。
6. 拒接: 被叫呼入, 有呼入提示后按开关机键拒绝接听该号码。
7. 紧急求助: 空闲状态下按下紧急求助按键则主叫呼出已设定的紧急求助号码, 通话结束后上传通话结束信息到设定的服务器 (详见通讯协议章程), 紧急求助按键为多功能按键, 长按 (超过 3 秒) 为紧急求助功能, 如果空闲状态下不是长按则视为无效按键 (误按)。
8. 急救服务: 长按 (超过 3 秒) 急救服务按键, 上传急救服务请求信息 (详见通讯协议章程) 到指定的服务器, 上传结束后挂机等待。
9. 日常服务: 长按 (超过 3 秒) 日常服务按键, 上传日常服务请求信息到指定的服务器, 上传结束后挂机等待。

10. 语音报时：按下语音报时按键即刻语音报出当前时间（24 小时制几点几分）。

11. 信息设置：可设置如下信息内容

- 1) 通讯方式设定（点对点/GPRS）。
- 2) 紧急求助呼出号码设置，紧急求助信息上传服务中心号码设置。
- 3) 急救服务信息上传号码设置。
- 4) 日常服务信息上传号码设置。
- 5) 允许号码信息列表设置，最多可设置 20 组。
- 6) 端机时间设置（主动请求）。
- 7) 语音报号开关设置。
- 8) 特服号码表设置，最多可设置 10 组。

12. 通话音量调节：

语音接通后可通过日常服务按键和急救服务按键来调节听筒音量大小。

短信要求的响应时间在 5 秒以内。

3.3 系统后台软件设计主要内容

后台软件主要完成在终端发出呼叫后，呼叫中心对终端请求的响应处理，包括语音请求和短信请求的信息接受、判断、转发、以及坐席的处理，异常监控，社区处理，服务机构处理，收费管理，系统管理等，具体结构如图 3.1 所示。

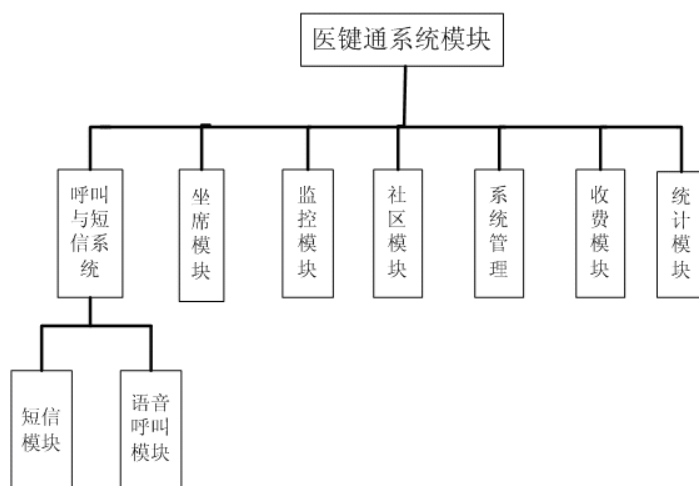


图 3.1 后台软件模块结构图

整个系统跨越了短信协议，PSTN 协议，UDP 协议，数据库协议，http 协议，所以组网方式比较复杂。在该部分，对系统性能有影响的，协议解析部分，我们

采用 C++,其余部分采用.NET 设计。

Microsoft 的.NET 框架是 Microsoft XML Web Services 平台, 它主要包括以下的 4 个部分:

1. 客户端。它指用于个人计算机以及移动和嵌入设备的各种操作系统。例如, Microsoft Windows XP、Windows XP Embedded 和 Windows CE. NET, 就是实现. NET 框架的新一代软件客户端。

2. 服务。这些服务允许应用程序相互共享数据, 并调用其他应用程序的功能, 而不必考虑这些应用程序是如何构建的、它们在何种操作系统或平台上运行以及使用何种设备访问它们等。

3. 服务器。服务器用于驻留和部署: XML Web Services 以及. NET 平台。Microsoft. NET Enterprise Server、Windows 2000 Server、Windows 2003 Server 系列提供了内置的安全性, 并支持 XML 和可缩放功能。

4. 程序员开发工具。这些工具用于在. NET 平台的核心上创建 XML Web Services。Microsoft 公司推出了 Visual Studio. NET, 它提供的框架可以采用 4 种核心语言进行应用程序开发: Microsoft Visual Basic、Microsoft Visual C++、Microsoft Visual C# 和 Microsoft. JScript。虽然 Visual FoxPro 7.0 不是 Visual Studio. NET 的一部分, 但仍然可以用来创建 XML Web Services, 因而也是一种有效的. NET 框架开发人员工具^[14]。

根据系统的整体需求, 设计出完整的网络结构如图 3.2 所示。

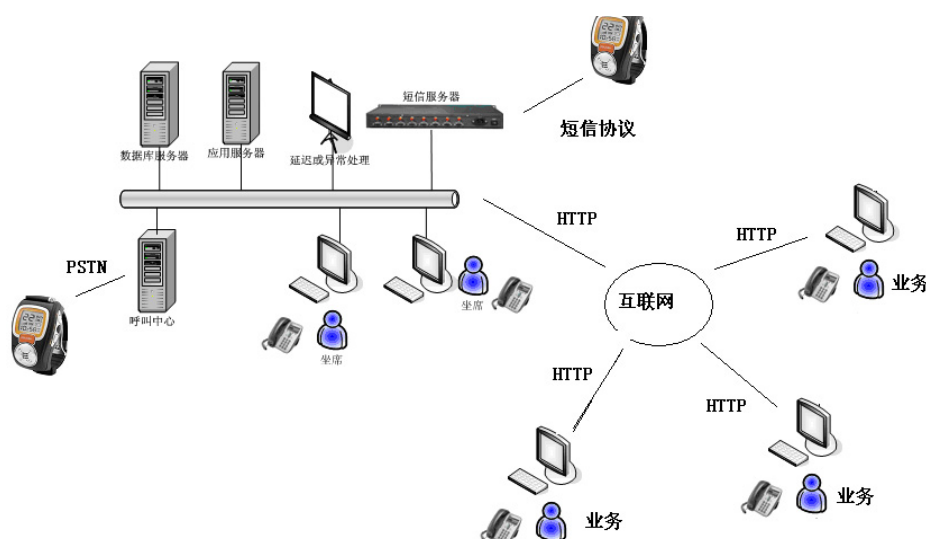


图 3.2 整体网络结构图

3.4 终端设计方案

终端平台采用 MTK 方案，采用 MTK2665 平台，该平台最大特点是价格便宜，硬件尺寸比较小，同时为最新芯片，在生产上能够得到保证。针对老年人视力问题，采用语音提示代替屏幕。所以整个终端只有 2 个 LED 灯，一个为红色表示需要充电，一个蓝色表示信号弱。紧急按钮为正上方，要求设计的比较明显，具有一定防震，防水效果，但是无法做到防水侵入效果。主要考虑设备大小的影响。最终外观如图 3.3 所示。



图 3.3 终端外观图

3.5 后台软件系统设计方案

本节定义了系统模块软件结构，并描述每个模块功能和内部组织结构。

3.5.1 软件系统功能划分

本系统一共两大子系统，服务器和系统业务模块。服务器子系统及时接收并及时处理用户信息，同时管理整个系统信息数据。系统业务模块是社区服务或指定的医疗服务机构对来自数据中心的请求进行处理，同时将处理的结果反馈到数据中心。

3.5.2 模块功能描述

1. 短信语音接收服务

本模块完成短信和语音求救两种不同的请求，对用户短信息的接收分析，其主要功能如下：接收和识别来自医键通用请求信息；过滤掉无用或垃圾信息；将识别后的有用信息提交至坐席服务；对于语音的服务，采用传统呼叫中心的平台，

通过网络协议或者中间件的技术，方便的得到语音请求的请求者的呼入号码，并按照对应的调度算法，把请求信息发送到坐席上。

2. 坐席服务

完成对服务请求者的电话第一次确认；用户终端在与坐席通话后，自动发送确认短信给工单业务处理模块，在这种情况下工单自动流转到下一阶段。这时坐席给用户亲属发送短信。将当前请求提交至监控坐席等待任务分配；处理社区服务人员提交的社区服务报告；电话确认该请求完成情况；审核并修改该任务单和社区服务报告状态；向亲属发送第二次服务结果短信息；将确认后的结果提交至监控坐席。

3. 社区模块

该服务的功能如下：为服务机构提供任务单据；将服务机构的社区服务报告提交至坐席。

4. 监控坐席

处理服务坐席提交后的任务请求；分配任务单；向亲属发送处理信息；实时监控当前系统任务调度和处理状况。

5. 统计模块

查询当前系统某个条件下的业务状况；统计当前系统某个条件下的业务状况；根据需要生成报表

6. 服务专席

根据服务坐席的请求信息，安排绿色通道；填写通道使用记录；电话通知用户；根据服务坐席反馈的信息将该用户请求归档。

7. 系统管理

添加系统资源（包括：业务坐席、专家、救护车、社区机构、用户等）；权限管理（角色管理和角色权限管理）；系统运行参数设置。

8. 社区服务机构

接收由监控坐席的任务单；完成对任务请求者的服务响应；向服务坐席报告当前请求完成情况。

9. 医键通终端

完成服务请求短信的发送；完成服务坐席确认通话；完成监控坐席的确认通话；完成服务专席的确认通话。

3.5.3 性能要求

1. 信息准确度

- 1) 请求信息要求准确，请求信息代码要求 4 位以上。

- 2) 服务描述信息要求准确简洁。
- 3) 完成情况描述信息要求准确简洁。

2. 时间特性

- 1) 服务资源信息要求及时更新。
- 2) 短信交互信息延迟要求不要超过 1 分钟。
- 3) 社区服务响应要求不超过 30 分钟。
- 4) 坐席服务响应时间不超过 5 分钟。
- 5) 声音要求 3 米外可听见。
- 6) 按键灵活，可靠性强。

3.5.4 系统用例图

用例(Use Case)是一种描述系统需求的方法，使用用例的方法来描述系统需求的过程就是用例建模。用例方法最早是由 Iva Jackboson 博士提出的，后来被综合到 UML 规范之中，成为一种标准化的需求表述体系。用例的使用在 RUP 中被推崇备至，整个 RUP 流程都被称作是“用例驱动”(Use-Case Driven)的，各种类型的开发活动包括项目管理、分析设计、测试、实现等都是以系统用例为主要输入工件，用例模型奠定了整个系统软件开发的基础[15]。

本部分主要通过用例图来描述系统各个部分的动态功能。图 3.4 为系统整体用例。

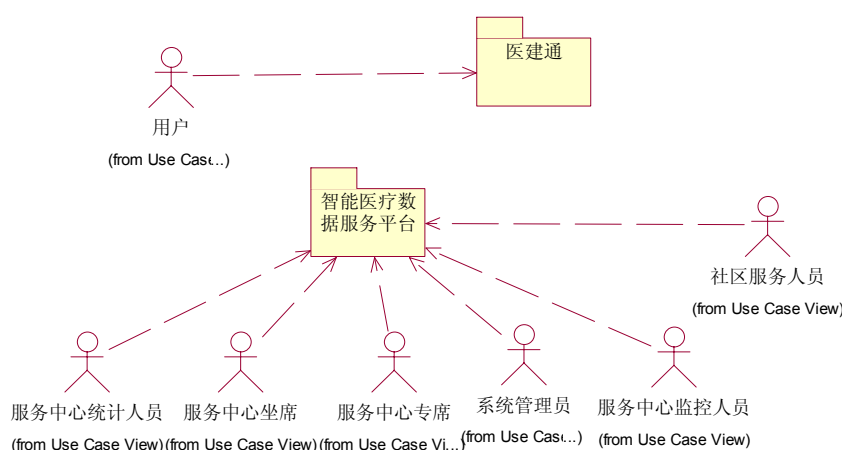


图 3.4 整体 use case 图

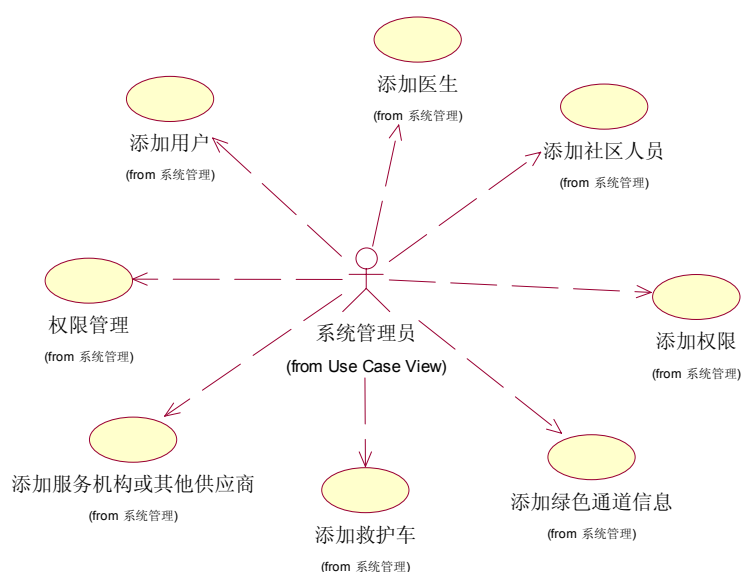


图 3.5 系统管理 use case 图

图 3.5 为系统管理，主要包括用户管理、权限管理、医生管理、社区人员管理、服务机构管理、救护车管理、绿色通道管理，在 UML 中，任何一张表都有 CRUD 四项功能，但是用例又必须是一个完整场景，这是一个自相矛盾的问题，所以在这个地方，UML 没有明显的要求，添加用户实际映射到用户管理，其他情况类同。

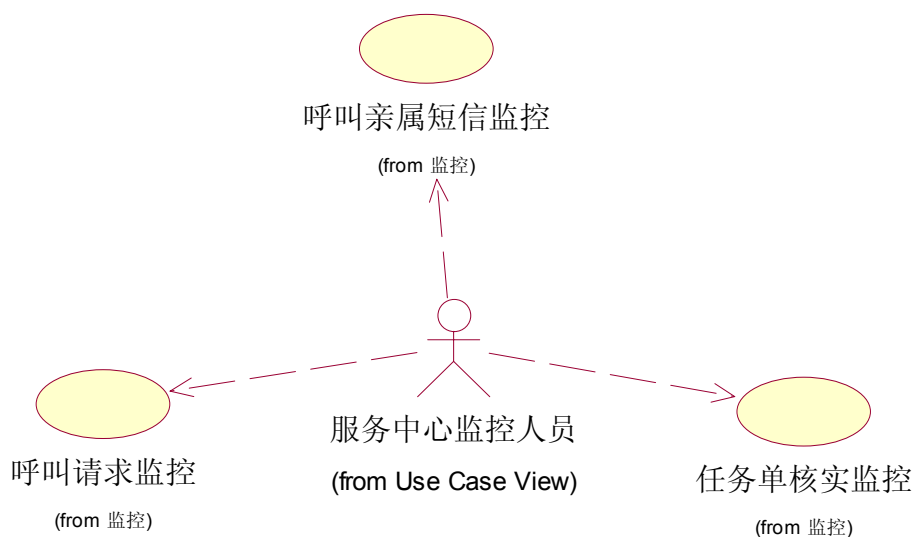


图 3.6 系统监控 use case 图

图 3.6 包括呼叫请求监控、呼叫亲属短信监控、任务单核实监控、主要完成对异常情况的监控。由服务中心的监控人员完成。问题解决不在系统管理范围内。

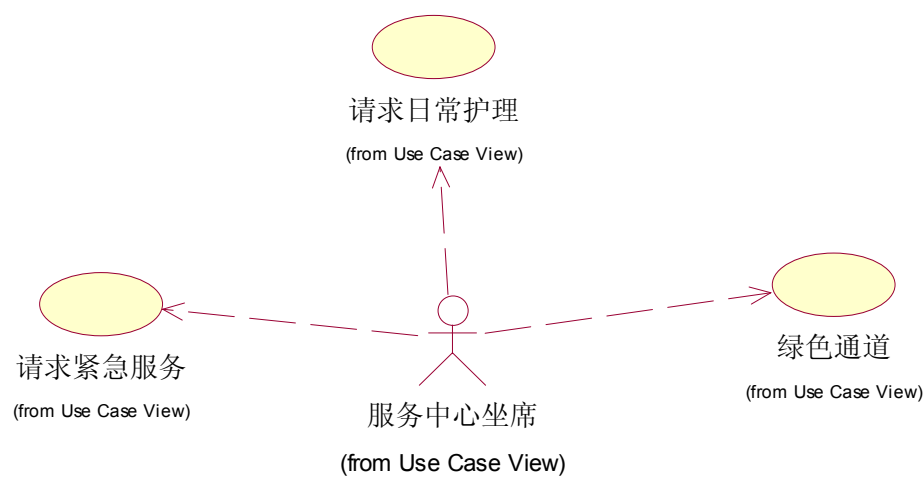


图 3.7 服务坐席 use case 图

坐席是本系统的重点，从图 3.7 可以看出，主要完成请求日常护理，请求紧急服务，绿色通道这几个主要功能。每个请求，都要分为请求阶段，请求确认阶段，任务分派阶段，任务核实阶段，最终归档。所有阶段要进入下一个阶段，都是由系统自动根据情况进行推动，只有在出现特殊情况才能由人工介入，例如用户刚开始是请求日常护理，处理过程病情加重，要进入紧急服务，这个时间，就需要改变原来的服务，需要人工干预，但是需要进行异常原因和操作记录，保证有证可循。这部分异常情况可能出现在任何阶段，都要能进行处理。

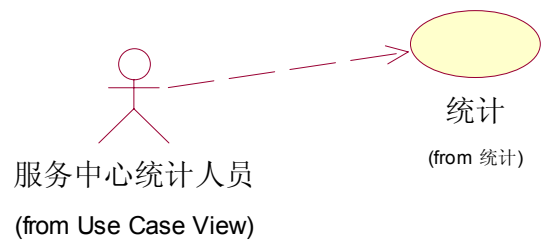


图 3.8 统计坐席 use case 图

图 3.8 说明了对坐席工作的统计情况，主要包括未正常处理的统计，处理时间的统计，处理数量统计，满意程度统计等。统计主要是为系统运营服务，使管理的重要内容，在以后系统运营过程还需要改变，这部分设计必须考虑到这个变化。

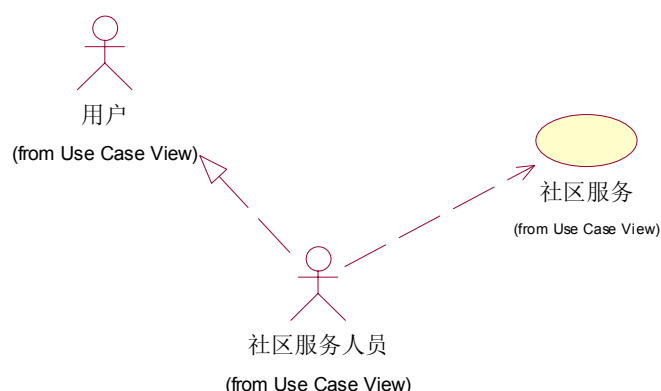


图 3.9 社区服务 use case 图

如图 3.9,社区服务主要是用来通知社区护理人员,派遣的任务单已经下达,社区人员需要填写处理报告,包括对社区人员不满意的处理警告,该部分采用自己提供的无风扇,无硬盘的终端设备。便于安装和施工,维护。

3.6 SaaS 模式的应用设计

传统的软件产品都是以项目为先导的,主要关心的是项目交付之前的工作,在项目交付之后,虽然也有维护在内,但是并不关心产品后续发展的问题。

这个带来几个方面的问题:

1. 对于软件产品的使用者提出了过高的要求,不仅仅要求使用者能够维护和使用软件产品的软硬件,而且解决运营过程的业务问题,这个对于像中国电信这样的企业没有问题,但中小企业就会遇到困难。
2. 对于传统企业,虽然具备运营能力,但是不了解软件产品经营的特点,使得很多服务无法进行;影响了软件产品的使用范围。
3. 软件项目可以分为服务型和生产型的,很多服务型的项目由于运营模式的问题,解决不了后续运行的费用持续投入,最终无法真正的投入使用。

如何利用传统企业的优势,使得本产品能够得到大范围的推广,使得更多用户能够享受到更好的医疗服务,是本次 SaaS 设计的重点。

同时,如何保证产品能够能够持续运行和发展,也是本次 SaaS 设计的重点。

在这个基础上,我们进行了长期的调研和论证,推出了基于 SaaS 的应用模式,从而保证运营企业的可持续发展。

如图 3.10 所示,系统主要从月租,增值服务和超时服务三个方面来完成持续

运行和发展的，月租是每个月的固定收入，传统固话收费模式已经深入人心，在这方面系统主要参考固话收费模式。



图 3.10 收费模式

增值服务是目前互联网采用的方式，例如 baidu,google 等，中国电信运营商的 112 和 12580 也是采用这样的方式，都取得了很大的成功；这些增值服务的收费模式完全超出传统经营模式的理念，例如陕西电信 114 中的搬家查询服务，每年的增值服务超过 300 万元，这些收费主要是商家而不是用户，所以这部分的重点主要在于研究那些业务可以提供更多的商家加入本系统。

超时服务是学租和增值的一个补充方案，防止用户无限制的使用系统资源，同时也能满足有的特别用户的特殊需求。

3.7 本章小结

本章主要完成了系统总体方案的设计；主要包括 5 个方面，第一部分确定了设计总体目标，第二，第三部分说明了终端系统主要设计内容，以及后台软件主要功能；第四和第五部分分别定义了终端和后台软件的设计方案，列出了系统需要达到的各项性能指标和详细的参数。

第四章 系统详细设计及实现

本章主要包括终端流程设计和后台软件详细设计两个部分。本章节详细说明了终端，后台各个模块，各个业务流程的详细设计；通过详细的流程图图说明终端的业务流程，通过类图和 MVC 方式描述了类。然后组织编码人员依照详细设计并进行编码实现，最后依照系统设计目标编写系统测试方案和实施方法，作为开发和验证人员的测试工作指导。

4.1 终端的详细设计

开机流程主要描述开机完成的主要动作和对事件的准备工作。除等待事件外，其他部分都由终端自动完成,具体如图 4.1 所示。

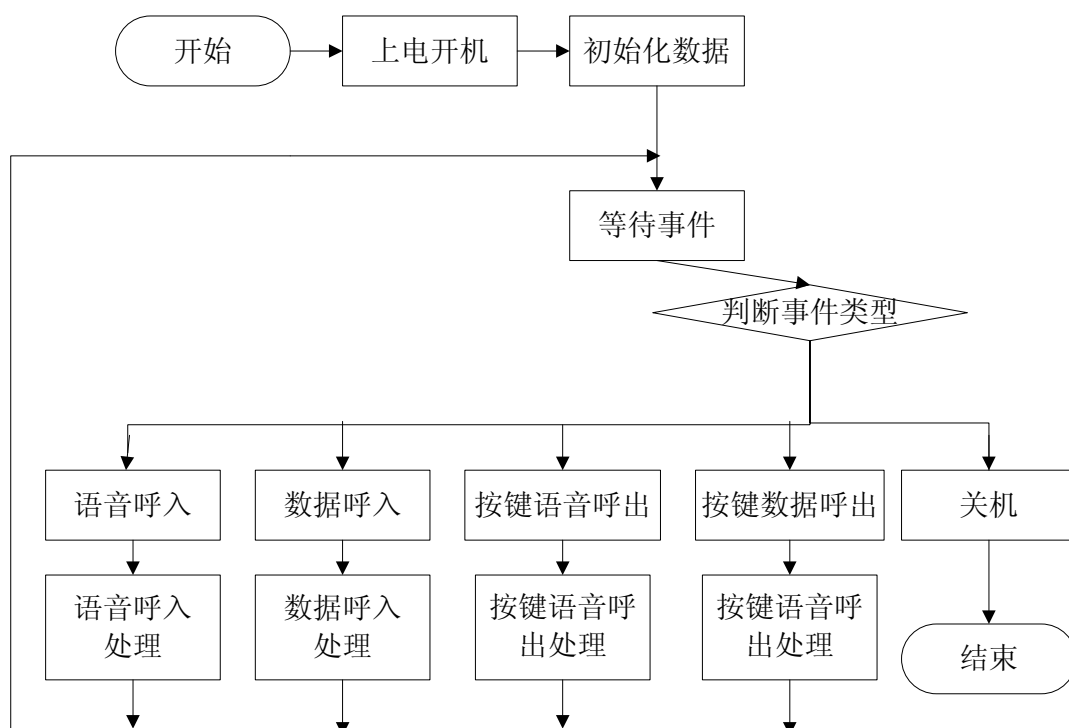


图 4.1 开机主流程图

语音呼入处理流程，主要说明接听电话时终端的过程，在此过程必须首先保证系统处于空闲状态，主要是不处于接听电话，或者正在报时，正在进行参数设

置等这些状态，用户若在允许接听号码表中设置电话号码，则不在此表中电话会被自动拒绝，若此表空，则自动接通，不需要用户按下接通按钮，并且在电话语音结束后会发送相关短信内容到服务器，具体流程如图 4.2 所示。

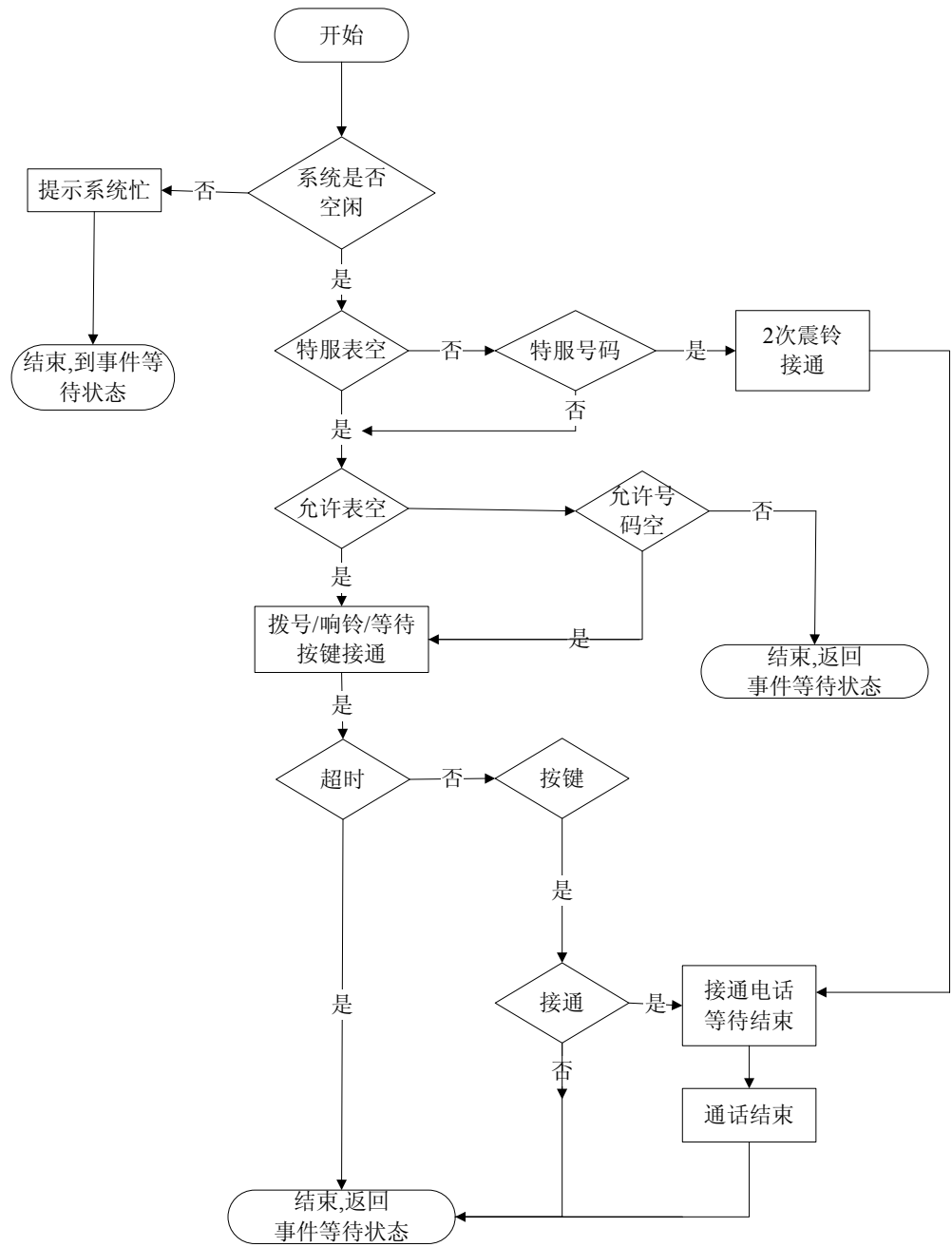


图 4.2 语音呼入处理流程流程图

数据呼入处理流程主要用来说明对终端的参数设置过程，这里面包括短信服务器号码设置，呼叫中心号码设置，允许接通号码设置，其中允许接通号码设置可以为空，具体流程如图 4.3 所示。

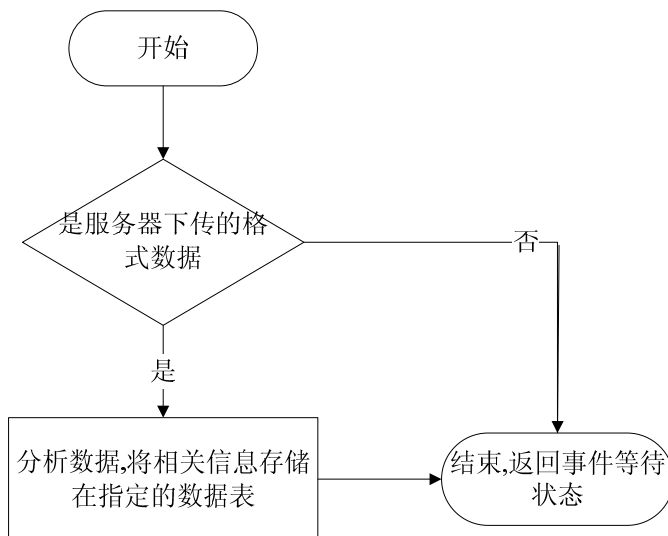


图 4.3 参数设置流程图

按键数据事件（报时按键），主要是当按下键能进行当前时间的语音报时，方便老年人使用，同时具备对时功能。当按下时间超过 3 秒，就看作启动对时功能，当系统设置为 gprs 模式则采用 gprs 进行数据通讯，否则使用短信方式，具体流程如图 4.4 所示。

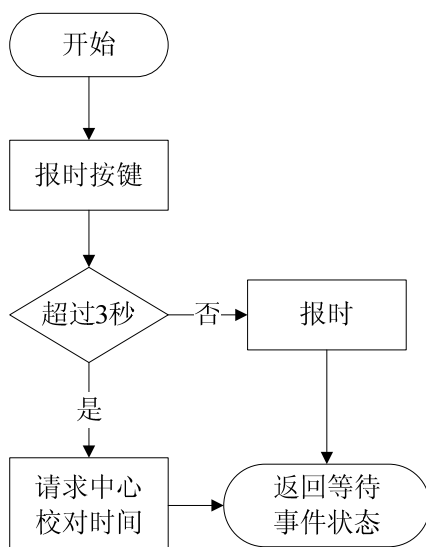


图 4.4 参数设置流程图

按键语音事件（紧急求助），当使用者在摔倒在家中或者户外，周围没有照顾人员，只要使用者按下这个按钮，则自动拨入到通过参数设置存储在设备上的呼叫中心的号码，无须这时一个一个数字进行拨电话号码，在通话结束会自动发送呼叫信息到短信服务器，具体流程如图 4.5 所示。

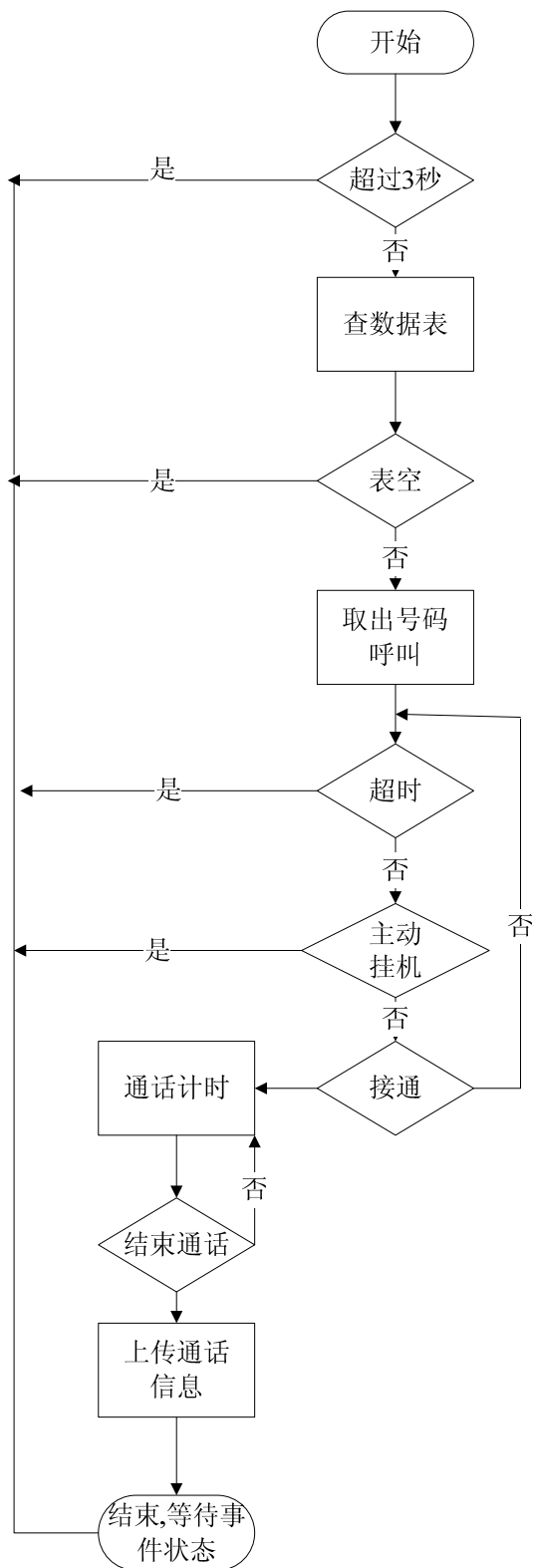


图 4.5 语音紧急求助流程图

按键数据事件（急救服务），这个事件是为了向后台传输按键的信息，通过参数设置的运行模式，分 gprs 和短信两种方式发送。后台服务器参数必须设定好，否则无法发送信息，具体流程图如图 4.6。

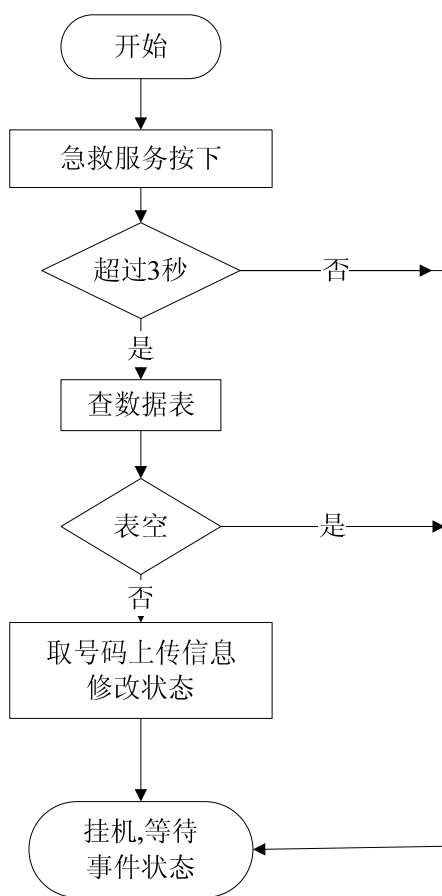


图 4.6 紧急数据求助流程图

紧急服务有时间的要求，当使用短信方式超过 1 分钟没有服务中心的电话打进来，需要使用紧急语音服务来完成相关操作，这部分也可以通过 GPRS 方式来完成，不会存在这样的问题。这种情况需要的服务不是危及生命的情况，例如在晚上发烧，或者发现高血压的药品没有了，这种比较重要的事情。当然用户也可以选择紧急语音请求，但是语音收费标准要高一些。主要是因为语音服务是同步的，会造成资源的浪费，同时不利于请求类型的分离。不容易提高处理效率。这个服务是目前系统比较界定模糊的一个服务。

按键数据事件（日常服务），这个事件与按键数据事件（急救服务）主要是请求服务的类型不同，在后台处理的紧急程度要低，其他和急救服务是完全类似的，他主要完成送药，家庭病床服务，协助挂号，联系专家等日常生活遇到的问题，甚至可以送米，油等。具体流程如图 4.7 所示。

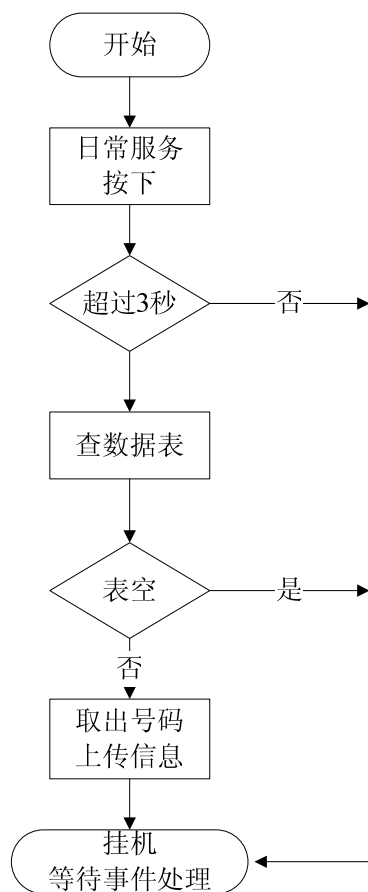


图4.7 紧急数据求助流程图

4.2 后台模块详细设计和实现

通过第三部分的分析，我们已经得出系统大部分的功能和用例，和对象，以及数据库信息，但是我们还不能直接进行编码，在详细设计阶段，我们将根据这些信息，对对象进行归类，找到类的接口，并对用例和类进行细化和优化，遵循设计的一些相关定理和基本原则，进行处理。我们采用 MVC 模式，把整个类分成界面类，操作类和实体类，同时把部分实体类转化成数据库类。这一部分也非常的重要，大部分的设计人员都会依据经验强行在头脑中先构造出模块和子模块的这样的关系图，这不符合面向对象的设计原则和方法，面向对象的最大特点是其方法坚持的逐步进化的特点，而不是人为的化分。

也许经验丰富的设计人员可以按照经验设计出非常完美的设计，但是其化费的时间，人力以及安全性很难得到保证。目前大部分的设计人员还在坚持这种方法，主要是受结构化程序设计的影响比较深，而且没有在设计中尝试到 OOA 的优势。本项目我们采用 OOA 的设计思路，并最终得到系统的详细设计。

4.3 详细描述概述

主要用静态特征和动态特征描述类。对于静态特征和动态特征的描述方法，选用 UML 标记方法。其中静态特征主要有类（包含类名，属性，方法），类结构图。动态特征主要有序列图，协作图。序列图和协作图是对同一过程的不同描述方法。序列图关心的是类（或对象）之间按时间顺序的互操作过程，协作图关心的是类（或对象）之间消息传递过程。

4.3.1 类的层次关系

对业务相关的类划分 4 个不同的层次：实体类，操作类，界面类，数据库类。

实体类的作用是将相关的数据进行聚合形成体现业务概念的业务实体并封装相应的操作。操作类是将实现相关业务功能的业务实体进行聚合并封装业务功能的实现。界面类是体现业务功能的表现形式^[17]。

我们通过对用例图序列图的分析，得出该序列图涉及到的业务对象模型和类图。其中类图主要涉及到界面，控制，操作类，实体类，而业务对象模型只涉及到操作类和实体类。

从业务对象模型去推倒数据库设计，对实体类进行组合和上卷与下拉设计，得出数据库的表类。最终得到整个系统的类层次图。

我们规定：界面类以 C 开头，操作类以 O 开头，实体类以 E 开头。表命名全部大写，但是具体编码部分可以建立灵活对应关系。不需要一定与设计一致。

以用户发出请求，坐席受理业务作为例子，来展示由那些类来相互协作完成这些功能，其中序列图略去，直接查看由序列图推导出来的坐席受理业务的类的关系图，并做了一个简单的优化，除了满足系统的功能之外，通过接口层分离了服务和实现部分，使得系统具有一定的灵活性，为后期的修改建立很好的架构基础。这个方法和过程，可以参考<<UML 数据库设计应用>>一书^[18]。

结合其他 UML 界面设计技术，我们通过对用例的各种描述，尤其是用例的叙述，用例的序列图，对象交互图等多种角度，从而得出对象及其交互，最终由对象得到的类图如下图 4.8 描述，在这里我们坚持 MVC 的模式进行基本划分，又采用前面的编码规则进行了命名。这些类在最后的实现得到了很好的体现。

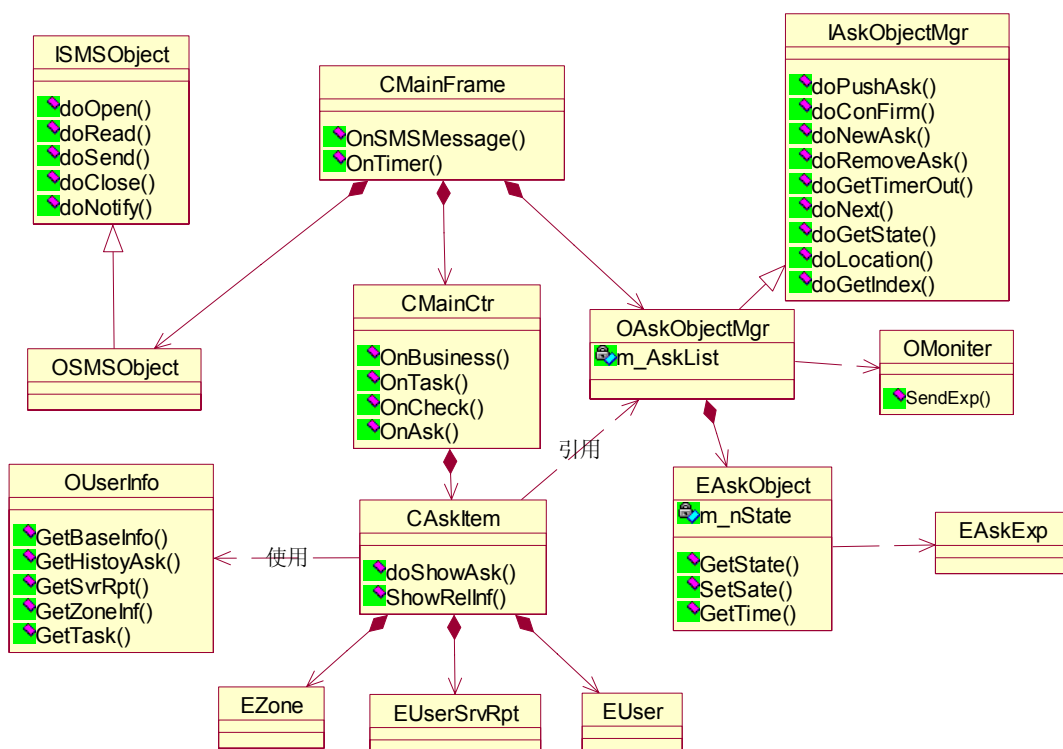


图 4.8 用户请求类图

这个类图我们在第三章可能得到了，当时还没有进行最后确定，但是如果在概要设计阶段工作做的细致和有效，这个阶段就不会出现很大差异，这个时间主要是接口函数的设计，我们一般不规定接口函数参数，流给程序员自己发挥。这样比较符合 RUP 的迭代开发，细节由程序员在每次迭代的编码阶段完成，经过本项目实验，基本能保证 80% 以上的设计不需要修改，仅仅需要细化。

同样，我们使用业务对象模型代替原来的 E-R 图表示方法，最后通过业务对象模型推导数据库详细设计，具体方法<<UML 数据库设计应用>>也描述的比较详细。当然，这部分在详细设计阶段已经存在，本阶段又进行了细化。业务对象模型^[18] 如图 4.9 所示：

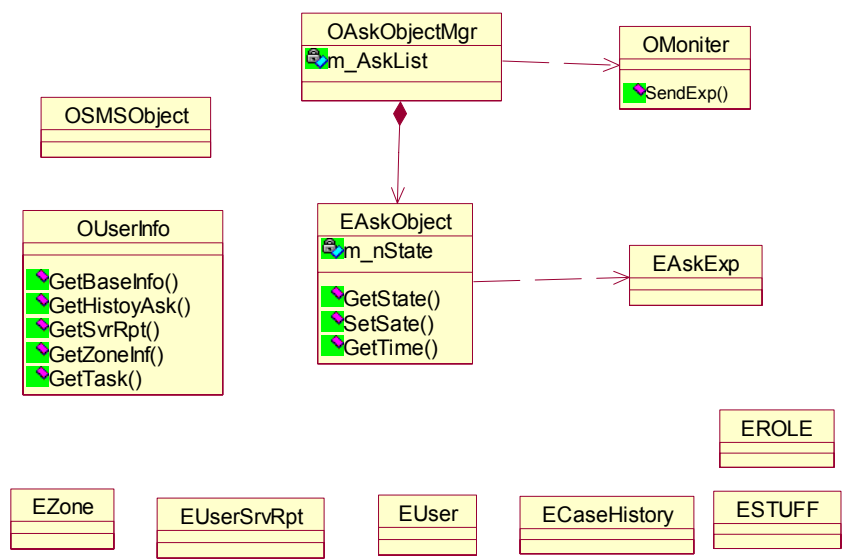


图 4.9 业务对象

每个用例都有一个业务对象模型，不需要一定要非常清楚描述其相互管理，但是希望能清晰描述其之间的业务关系，业务对象模型是设计数据库的基础，这些对象或功能加强，进化成类，或功能衰减变成数据库的表，当然也可能类之间发生合并，上卷，下卷，表也会出现这样的情况。

最终，通过以上方法，我们设计出了整个系统的三层的类，以及类之间的关系，下面是系统部分类的描述，也是本阶段的主要成果。

4.3.2 实体类

实体类主要是作为数据管理和业务逻辑处理层面上存在的类别；它们主要在分析阶段区分 实体类的主要职责是存储和管理系统内部的信息，它也可以有行为，甚至很复杂的行为，但这些行为必须与它所代表的实体对象密切相关^[19]，具有业务意义的最小操作单元。

表 4.1 系统实体表

| 编号 | 实体类名称 | 中文注解 |
|----|---------------|-----------------------|
| 1 | EResource | 服务资源 |
| 2 | EQuickServ | 绿色通道(便捷服务 convenient) |
| 3 | EServerVendor | 服务提供商 |
| 4 | ETask | 任务单 |
| 5 | ETaskReport | 任务报告(社区服务报告) |
| 6 | EUserInfo | 用户信息 |

| | | |
|----|---------------|---------------|
| 7 | EAsk | 用户请求 |
| 8 | ENetWorkAgent | 网络通讯代理类 |
| 9 | EClientSocket | 客户 socket 管理类 |
| 10 | ETaskStrage | 坐席决策调度类 |
| 11 | EZone | 社区 |
| 12 | ETaskObject | 任务对象 |
| 13 | ECaseHistroy | 病历 |
| 14 | E User | 用户 |
| 15 | E Zone | 社区 |
| 16 | EuserSrvRpt | 服务 |
| 17 | EaskExp | 请求异常 |
| 18 | ETaskExp | 任务异常 |

表 4.1 是 18 个实体类，主要完成对数据库数据的操作和记录当前记录的值，在我们的项目中，我们发现这样的好处就是以后数据库设计改变了，只需要修改实体类，程序代码改动比较小。而且真正操作用存储过程来完成，这样灵活性还可以大大提高。有了这些实体类和业务对象模型，数据库设计人员就可以很快进行数据库设计了，关于权限管理没有单独拿出来讨论，因为这个是比较成熟和固定，所以实际上的实体类数量还是比较多的。

4.3.3 数据库设计

根据之前的需求分析把日常考核的实际业务逻辑抽离出来，从而变成与数据库表对应的表结构，建立数据库表之间的关系，形成目标责任制考核系统的数据库设计图，系统中数据库设计如图 4.10 所示。



图 4.10 任务与机构表图

图 4.11 描述了 3 张表，其中部门表是对整个系统使用单位的描述，任务派遣表主要描述任务的派遣关系，并不描述具体派遣内容，机构表主要描述合作单位，这些数据中存储合作单位电话号码等联系方式，以及能够提供的服务容量。

| 救护车信息 | 社区基本信息 | 处理报告 |
|---|--|---|
| 序号 int 资源名称 varchar(25) 员工序号 int 司机 varchar(25) 联系电话 varchar(25) 备注 varchar(50) 状态 救护车编号 varchar(25) | 社区序号 int 社区编号 varchar(100) 社区名称 varchar(25) 地址 varchar(100) 联系人 varchar(25) 电话 varchar(25) 人员数量 int 注册用户数 int | 任务序号 varchar(20) 社区序号 int 报告序号 int 详细描述 varchar(200) 填写人编号 int 审核人员编号 int 状态 int 报告编号 varchar(200) |

图 4.11 基本信息，处理报告与救护车信息图

图 4.11 中，救护车信息描述救护车的联系方式和使用状态，社区基础信息主要描述社区基本信息，是这些表中最重要的一张表，处理报告描述处理结果，以及处理人员，也是系统中最重要的一张表。

| 专家基本信息 | 其他服务机构 | 资源类型 |
|---|--|-------------------------------------|
| 专家序号 int 专家名称 varchar(50) 专业 varchar(50) 联系电话 varchar(25) 备注 varchar(200) 状态 int | 序号 int 名称 varchar(50) 服务说明 varchar(100) 联系电话 varchar(25) 备注 varchar(200) 状态 int | 序号 int 名称 varchar(100) 状态 int |

图 4.12 专家信息与服务机构信息

图 4.12 中专家信息主要提供咨询专家的联系方式，帮助老年人解决生活中遇到的健康咨询问题。其他服务机构主要完成送米，面，油等生活用品的公司，和医疗没太大的关系，资源类型主要是个分类和说明表，便于统计和显示，是以后统计需要，尤其是外聘的管理，目前各家医疗机构都在进行这方面的管理，这是我们以后要努力和加强的地方，目前先满足简单的应用。

图 4.13 主要包括 2 部分，是权限管理，权限管理包括授权和日志管理，同时能提供针对不同用户登陆时显示的界面内容进行控制。

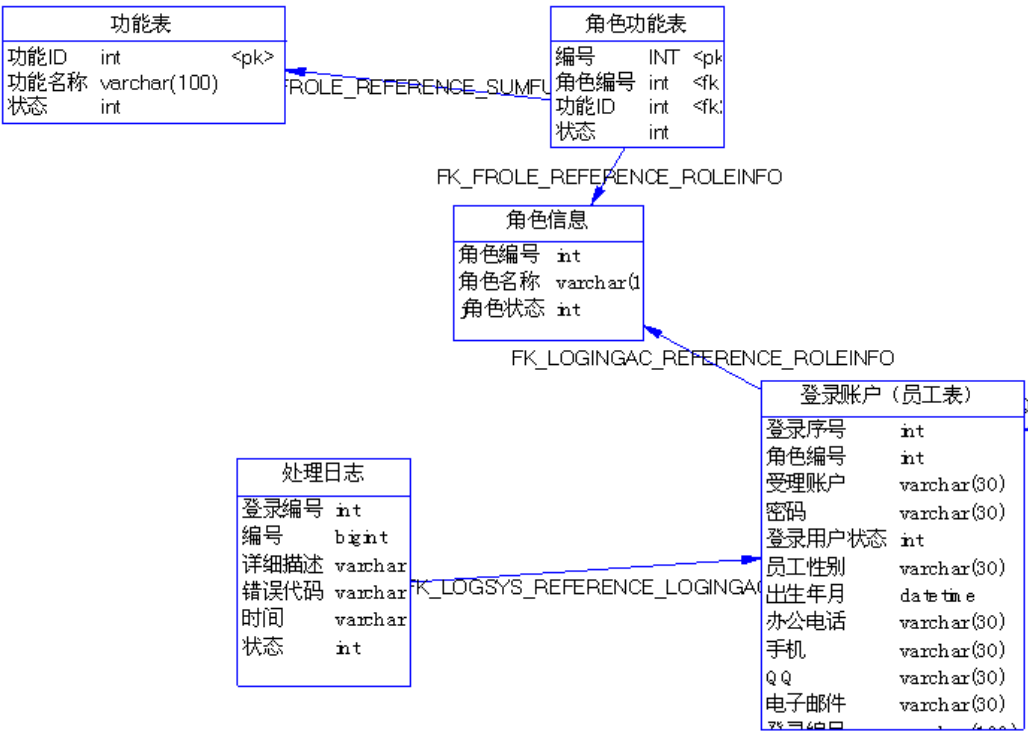


图 4.13 权限管理图

任务管理如图 4.14，主要包括请求短信，任务单，短信类型，处理结果，任务处理状态等表，其中请求短信和任务单，处理结果是核心表，其他表是为这几张表服务的。

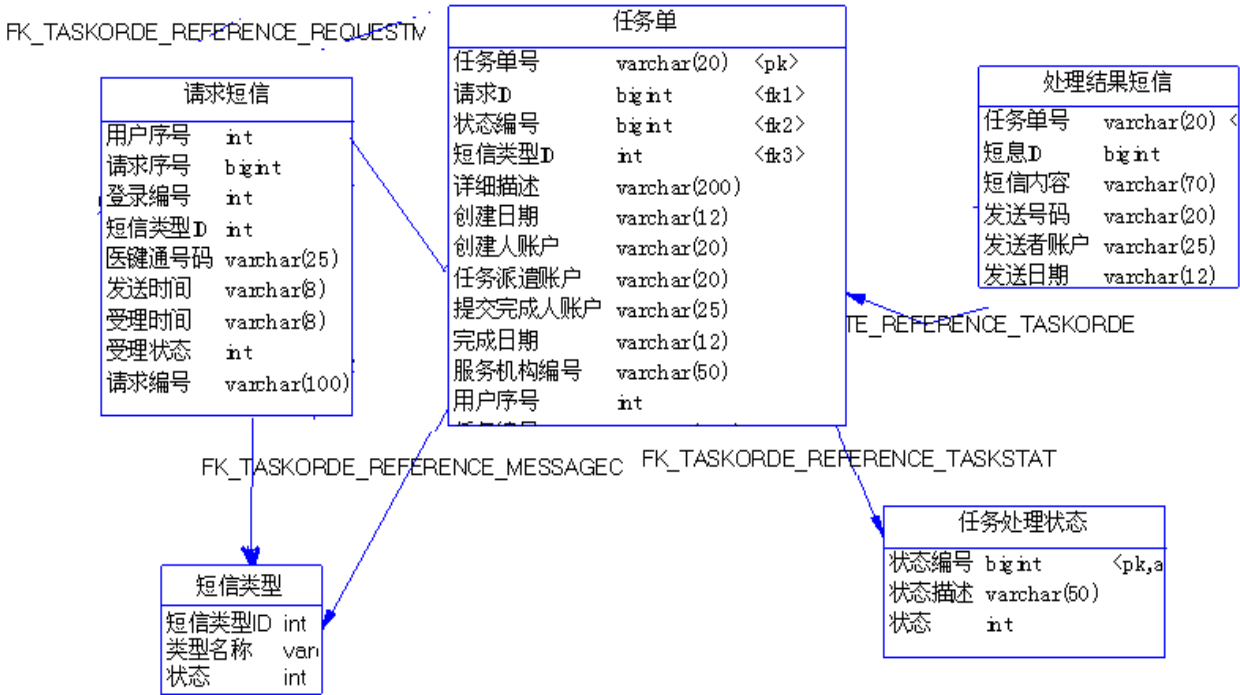


图 4.14 任务单与处理结果图

请求短信表主要记录短信的请求信息，包括类型，内容，短信发出人，发出时间，受理时间等，是系统任务单的驱动源泉，任务单主要描述用户请求派遣到哪个服务机构，以及任务单生成的时间，经手人员，任务单的详细描述。这部分比较复杂，可能回出现任务单的部分内容重新修改。要求系统设计能应付这种情况。

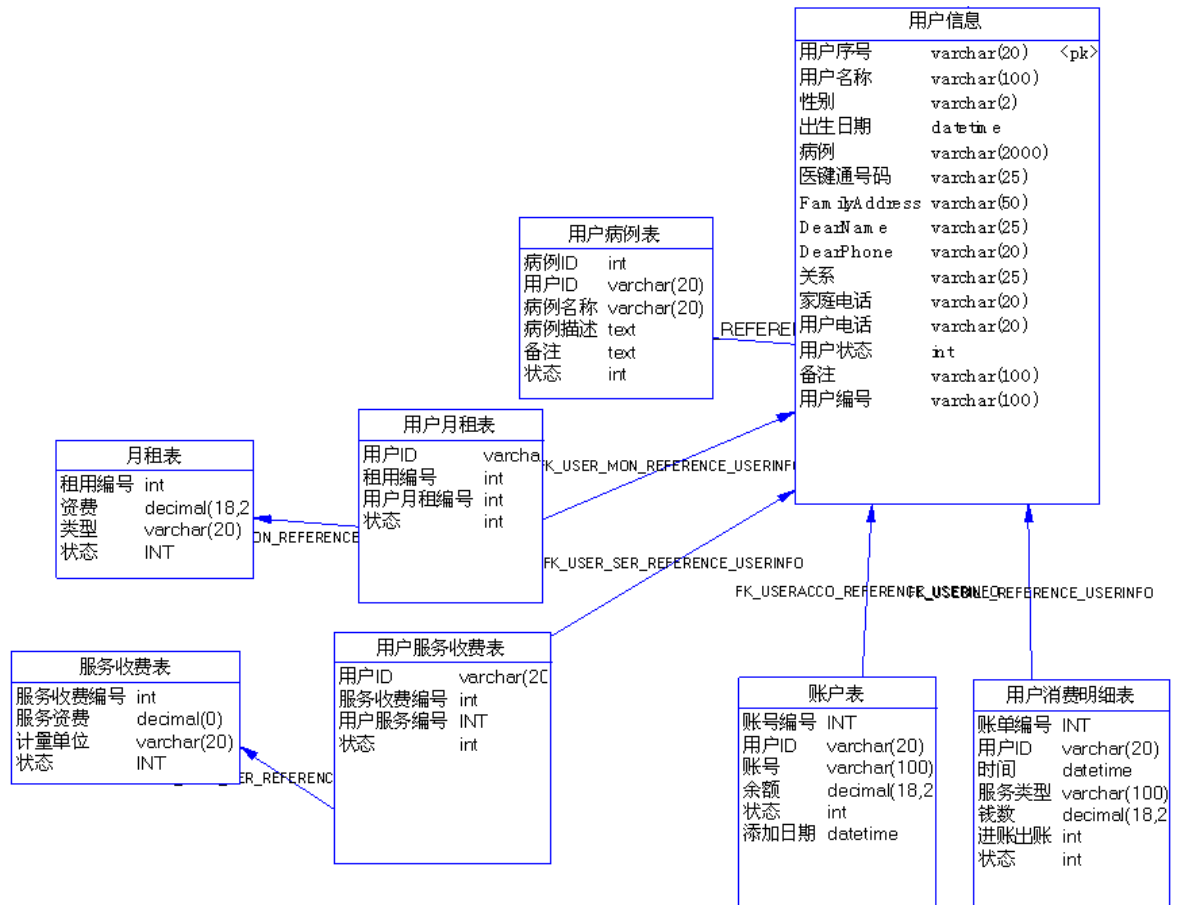


图 4.15 用户信息相关表图

图 4.15 主要包括系统收费管理的一些表，主要完成费用收取和帐务明细处理，月租主要记录月租类型和费用，用户信息记录用户的详细信息，病例记录用户的病例内容，收费表明收取费用的情况，消费明细记录用户付费的流水帐，帐户是用户缴费的标识，收费内容包括月租和其他费用，例如救护车，送药等。

| 请求异常表 | | 任务处理异常表 | |
|--------|---------|---------|---------|
| 用户ID | varchar | 任务单号 | varchar |
| 请求ID | bigint | 请求ID | bigint |
| 登录编号 | int | 状态编号 | bigint |
| 短信类型ID | int | 短信类型ID | int |
| 医键通号码 | varchar | 详细描述 | varchar |
| 发送时间 | varchar | 创建日期 | varchar |
| 受理时间 | varchar | 创建人账户 | varchar |
| 受理状态 | int | 任务派遣账户 | varchar |
| | | 提交完成人账户 | varchar |
| | | 完成日期 | varchar |
| | | 服务机构编号 | varchar |
| | | 用户ID | <Unde |

图 4.16 异常表

图 4.16 记录的是所有的异常信息，根据这些信息，将决定最终的业务流向，这些信息不是错误信息，是事情发生的情况与主业务正常流程不太相同。也是业务的一种形式。

4.3.4 操作类

将实现相关业务功能的业务实体进行聚合并封装业务功能的实现，是连接实体类和界面类的桥梁^[19]。

表 4.2 系统操作类表

| 编号 | 操作类名称 | 中文注解 |
|----|-----------------|------------|
| 19 | O_MessageAgent | 短信代理（服务器方） |
| 20 | O_Mointer | 服务监控 |
| 21 | O_AskObjectMgr | 请求对象管理者 |
| 22 | O_TaskObjectMgr | 任务对象管理者 |
| 23 | O_SMSObject | 短信代理（坐席） |
| 26 | O_UserInfo | 用户相关信息 |
| 27 | OTaskObjectMgr | 任务管理 |
| 28 | OExpAsk | 异常请求处理 |
| 29 | OExpTask | 异常任务处理 |
| 30 | OAccount | 帐务处理 |
| 31 | OpayMent | 缴款处理 |
| 32 | OServStatic | 服务统计 |

操作类主要完成系统的业务控制，其中短信服务器（O_MessageAgent），请求对象管理（O_AskObjectMgr），任务对象管理（OTaskObjectMgr），服务统计

(OServStatic)是系统的核心操作类,其他操作类功能相对比较简单。这些操作类接收界面执行或者由短信自动触发,在实体类的协助下进行相关业务处理。并把处理结果反馈给界面类。

4.3.5 界面类

实现业务功能的表现形式,界面类通过显示的形式,把信息转达给用户。不同界面类通过不同的显示,来表达模型的数据和状态信息。界面类获得来自操作类的数据值,并用它们来更新显示。

表 4.3 系统界面类表

| 编号 | 界面类名称 | 中文注解 |
|----|--------------------|--------|
| 33 | CMainFrame | 坐席主界面 |
| 34 | CMainCtr | 主控制界面 |
| 35 | CAskCtr | 请求控制 |
| 36 | CTaskCtr | 任务控制 |
| 37 | CCheckCtr | 审核控制 |
| 38 | CFinishedCtr | 归档控制 |
| 39 | CAskItem | 请求面板 |
| 40 | CTaskItem | 任务面板 |
| 41 | CCheckItem | 审核面板 |
| 42 | CFinishedItem | 归档面板 |
| 43 | CPayItem | 负费面板 |
| 44 | CAccoutItem | 帐务面板 |
| 45 | CStatic | 统计主界面 |
| 46 | CStaticCtr | 统计控制 |
| 47 | CStaticItem | 坐席统计面板 |
| 48 | CUserStaticItem | 用户统计面板 |
| 49 | CZoneStaticItem | 社区统计面板 |
| 50 | CServStaticItem | 服务统计面板 |
| 51 | CAccountStaticItem | 帐户统计面板 |
| 52 | CPayStaticItem | 付款统计面板 |
| 53 | CSysCustom | 客户信息 |
| 54 | CSysRole | 权限 |
| 55 | CsysStuff | 员工 |

| | | |
|----|------------|-------|
| 56 | CSysIP | IP 配置 |
| 57 | CSysZone | 社区 |
| 58 | CSysDepart | 部门 |
| 59 | CMonthRent | 月租 |
| 60 | CPayType | 付费类型 |
| 61 | CServUnit | 服务单位 |

从 33-61 是所有设计中的界面类，这些类定义了所有界面的框架，特性，包括接口，他们与操作类协助，完成系统的功能，界面类全部从系统类库继承而来，在实际编程过程中如果遇到困难，可以考虑使用汇入方式来完成相关操作，有关相关技术可以参考 GOF<<设计模式>>中相关解释。

表 4.4 短信服务器类表

| 编号 | 操作类名称 | 中文注解 |
|----|-----------------|----------|
| 62 | ISMAagent | 短信收发类接口 |
| 63 | ISite | 坐席接口 |
| 64 | OSMSimulate | 测试用短信收发类 |
| 65 | OSMWorkstation | 短信猫 |
| 66 | ISMSTDispatch | 调度接口 |
| 67 | OCircleDispatch | 循环调度 |
| 68 | IStrategy | 算法接口 |
| 69 | OShiftStrategy | 轮询算法 |
| 70 | OMedicalSite | 医疗坐席 |
| 71 | CSocket | 微软库 |
| 72 | CSerial | 第三方库 |

短信服务器主要完成短信收发处理，其中循环调度和轮询算法，医疗坐席这几个类是重点，其他是服务完成调度工作，以上为系统主要类，在编码过程还会增加 1 半以上。有关类结构图描述在此省略。

4.4 短信服务器代码分析

短信服务器服务器主要完成语音，短信的接收和输出，并且把收到的短信按照一定的要求进行格式化，在此基础上调用合适的调度算法，分配到终端，其中短信格式在不同阶段采用不同的接入方式可能不同，对于终端坐席发送的短信，

收到指令<CRLF>+CMTI:" SM" ,INDEX(信息存储位置)<CRLF>。

PC 上的控制软件通过读取 PDU 数据的 AT 命令: AT+CMGR=INDEX<CRLF>

将 TC35 模块中 PDU 格式的短消息内容读出。如果用+CMGL 代替+CMGR, 则可一次性读出全部短消息。

通过 TC35 删除短消息的方法为:

PC 上的控制软件收到一条短消息并处理后, 需要将其在 SIM 卡上删除, 以防止 SIM 卡饱和。删除短消息的指令为: AT+CMGD=INDEX<CR> 删除后模块返回<CRLF>OK<CRLF>

4.5 编码实现

在完成详细设计后, 模块具体的编码工作由编码人员按照详细设计说明实现, 程序采用 C++和 C#语言, 开发环境采用 VS2005, 编码总共进行了 4 个月, 6*30 人天。各个模块负责人依照笔者编写的测试方案进行详细测试规范制定和验证测试工作。

由于代码比较多, 同时代码实现也不是本论文的重点, 所以下面挑出 OSMWorkstation 类的实现部分, 进行分析和说明。

```
class OSMWorkstation :public ISMAgent
{
public:
    virtual int  doRead(char *pReadBuff) ;           //读取短信内容
    virtual bool doWrite(unsigned char *pReadBuff,int len) ; //发送短信
    virtual void doNotify() ;                       //短信来到
    virtual bool doOpen(char *Dev,int port,int flag); //打开设备
    virtual void doClose();                          //关闭设备
    bool      gsmSendMessage(const SM_PARAM *pSrc ); //发送短信
    int       gsmReadMessage(SM_PARAM* pMsg);        //读取短信
    bool      gsmDeleteMessage(const int index);      //删除短消息
    // UCS2 解码 返回: 目标字符串长度
    int gsmDecodeUcs2(const unsigned char *pSrc, char *pDst,int nSrcLength) ;
    //可打印字符串转换为字节数据 返回: 目标数据长度
    int gsmString2Bytes(const char *pSrc, unsigned char *pDst,int nSrcLength);
    // 字节数据转换为可打印字符串 返回: 目标字符串长度
    int gsmBytes2String(const unsigned char *pSrc, char *pDst, int nSrcLength);
private:
    CSerial m_serialPort;           //串口对象
    SM_PARAM m_smParam[100];       //短信缓冲区
    int m_iCurrRecord;              //当前记录
    int m_iCount;                  //目前记录数量
};
```

这个类完成完成真正的短信业务处理，但是由于短信可以通过很多方式进行处理，所以定义了 ISMAgent 规范了它的接口，以便以后在使用不同的处理实现时，不会造成代码的修改，从而做到实现依赖接口，并且符合替换原则和开环-闭环原则这样面向对象的设计的基本原则，来强化程序的灵活性。

```
#ifndef INTERfACE_H
#define INTERfACE_H
//短信代理类
class ISMAgent
{
public:
    virtual int  doRead(char *pReadBuff) = 0;           //读取短信内容
    virtual bool doWrite(unsigned char *pReadBuff,int len) = 0; //发送短信
    virtual void doNotify() = 0;                       //短信来到
};
```

同时把ISMAgent写成抽象类，也保证了接口类不会被实例化，从而避免在程序设计时没有使用实现部分的代码，进行自动的代码检测。

4.6 本章小结

本章详细介绍了笔者负责的系统详细设计和实现流程

第一部分详细介绍终端设备的各个键盘功能流程。第二部分描述了模块的详细设计与实现。第三部分描述了软件平台涉及到的类的详细设计。第四部分对短信服务器代码进行分析，最后给出实现部分细节。

至此，本次设计的课题已经完成，在分析基础上，确定了系统设计目标和总体设计思路 and 具体系统设计方案，然后对笔者负责的架构进行详细设计和实现。

在项目团队其它设计开发人员共同努力下，编码开发工作已经完成，目前正在利用测试工具进行模块自测，近期将进入到系统联调和测试阶段，预计将按照原计划在 4 月份实现整个项目。

第五章 系统测试

本章将在前面几章的基础上。对系统的功能和性能进行相应的测试，给出测试数据结果和说明。软件测试，就是一个过程或一系列过程，用来确认计算机代码完成了其应该完成的功能，不执行其不该有的操作^[20]。

5.1 测试环境

本节首先说明了测试所需要的测试环境，从硬件环境已经软件环境两个方面进行阐述。然后重点说明了测试方案，包括主要内容，详细的测试方法等，作为开发人员和集成测试人员进行系统联调测试依据。

测试环境组网

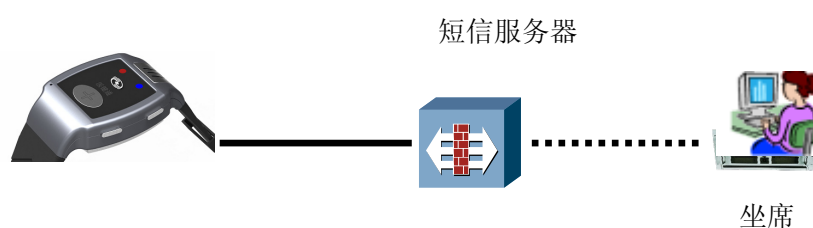


图 5.1 协议测试组网图

如图 4.17，项目做了大重的测试，自先使用手机对短信服务器进行测试，手机种类包括诺基亚，三星等大品牌，也测试了很多山寨手机的短信服务器接收能力，短信协议包括 PDU，TXT 等三种格式，目前服务器只能识别 PDU 格式。基本能满足 1 分钟内完成接收。

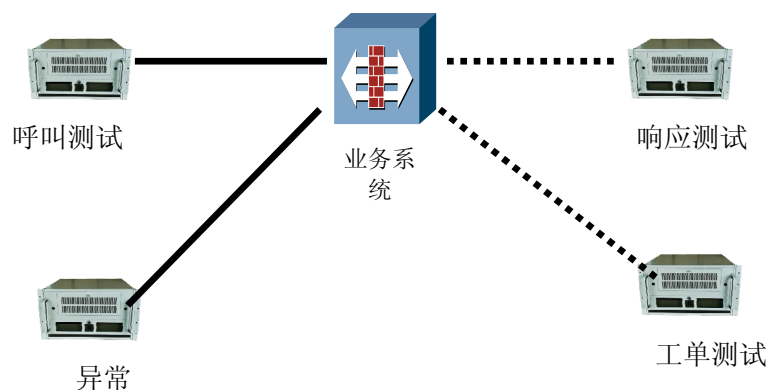


图 5.2 功能测试

同时我们也进行了系统功能测试，功能测试完全按照需求分析描述的进行测试，总共进行了大约 30 多次测试，每次 1 天时间，发现了很多异常问题，系统主要修改体现在异常情况，例如用户请求了服务，又要求取消，或者改变服务类型。

表 5.1 测试环境

| | |
|---------|-----------------|
| 数据库服务器 | 1 台 |
| 应用服务器 | 1 台 |
| 域名一个 | 为了 B/S 程序访问 |
| 短信服务器 | 1 台 |
| 路由器 | 1 台 |
| 医键通医疗平台 | 有服务器和客户端两部分 |
| 工控机 | 1 台 安装呼叫中心服务器软件 |
| 电话交换电路 | 1 块, 实现电话接入和呼叫 |
| 网络协议接口 | 软件与医疗服务平台进行通讯 |
| 电话 | 若干,与坐席人员相同 |

5.2 测试方案

5.2.1 功能测试

该小节主要完成对系统功能的测试，测试的功能包括：座席登录登出；拨号呼叫、短信呼叫、工单测试、异常测试，亲属短信发送，处理报告等。该部分的测试采用黑盒测试法，由人工操作完成。呼叫中心采用 COM 接口测试，短信服务器采用短信猫接口。测试结果见表 5.2。

表 5.2 功能测试结果表

| 功能 | 测试用例操作描述 | 测试结果 | 是否合格 |
|------------|-----------------------------------|---------------------------------------|------|
| 登录 | 用户座席 301 输入信息登录 | 登录成功 | 是 |
| 登出 | 用户座席 301 登出座席 | 登出成功 | 是 |
| 语音拨号 呼叫 | 用户终端 13572551933 呼叫中心 88453492 | 拨号成功，有回铃音，对方接听，进入通话状态;通话结束会自动向服务器发送短信 | 是 |
| 短信呼叫 | 用户终端 13572551933 呼叫中心 88453492 | 短信服务器接收到请求短信 | 是 |
| 确认短信 | 座席向终端拨打 | 能正常通话，通话后可自动发送确认短信 | 是 |
| 向亲属发 | 座席 301 填写短信内容，按下发 | 关联的系统数据库中的手机 | 是 |

| | | | |
|---------|--------------------------------------|--------------------------|---|
| 送短信 | 送短信 | 可以接收到短信 | |
| 服务商填写报告 | 服务商在收到工单后，向用户拨打电话，电话结束后进入报告填写界面，提交报告 | 座席 301 能够看到派遣的任务单的报告填写完毕 | 是 |
| 异常测试 1 | 坐席在收到用户请求 5 秒没进行电话询问 | 任务单变成红色，并且监控坐席收到请求异常 | 是 |
| 异常测试 2 | 服务商在收到用派遣单 5 秒没进行电话询问 | 派遣单变成红色，并且监控坐席收到请求异常 | 是 |
| 统计结果 | 系统按请求异常，工单处理异常，回访异常进行统计 | 能够显示相关信息 | 是 |
| 收费管理 | 系统能进行收费，扣除费用等 | 收费正常，扣费正常 | 是 |

5.2.2 性能测试

表 5.3 性能测试结果表

| | 性能测试描述 | 测试结果 | 是否合格 |
|-----------------------|----------------|------------------|------|
| 呼叫处理性能测试 | 1 分钟 1000 次呼叫 | 可以满足 | 是 |
| 信号强度测试 | 在户外，在室内，在常见场所 | 达到手机相同强度 | 是 |
| 短信交互信息延迟要求 | 不要超过 1 分钟 | 测试 100 次 不超过 5 秒 | 是 |
| 社区服务响应要求 不超过 30 分钟 | 1 分钟同时出现 3 次请求 | 测试可在 30 分钟完成 | 是 |
| 坐席服务响应时间 不超过 5 分钟 | 1 分钟同时出现 3 次请求 | 能满足要求 | 是 |
| 声音测试 | 3 米外能听见 | 能满足要求 | 是 |
| 语音拨号按键 | 不够灵活 | 需要改进 | 否 |

5.2.3 测试结果

由于系统界面比较多，下面用请求业务响应界面为例描述测试结果，图 5.3 为正常处理流程。

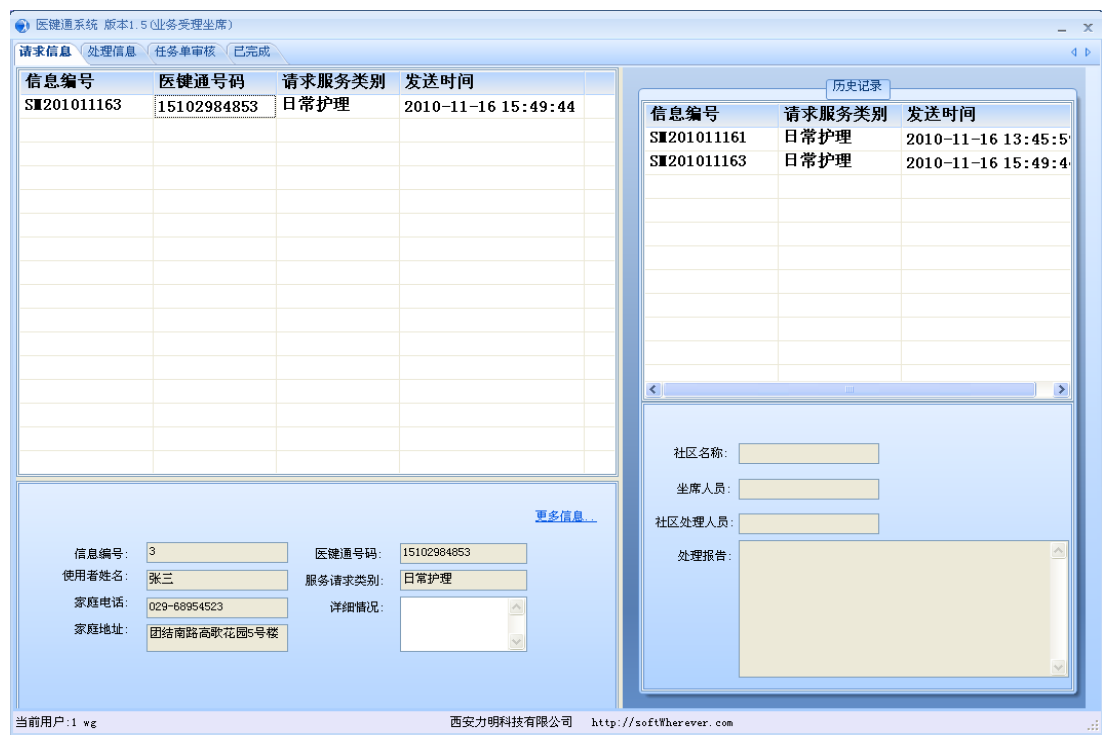


图 5.3 用户请求正常处理界面

当用户请求在规定时间内得到响应，例如 30 秒内，则系统回修改请求状态为请求得到响应状态，显示结果如图 5.3,具体过程如图 5.4:

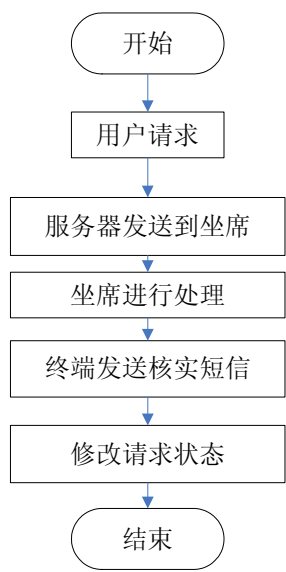


图 5.4 用户请求正常响应流程

5.3 小结

本章首先进行测试环境和测试方案的描述，接着对系统各个方面进行测试，给出了终端测试，性能测试，功能测试等方面测试内容，由于测试管理部分的人员变动，造成资料不足，没有给出确切的测试过程。

第六章 结束语

6.1 工作总结

从公司开始立项进行设计开发工作，到完成设备总体设计和所有模块的详细设计，在这段时间，设计团队细致认真研究了各项相关技术规范，分析设计架构，巧妙利用系统平台和软件架构，吸收了公司内外部很多先进设计方法，最终按时完成了设计方案。

经过 20 个月左右的辛勤工作，项目攻关组首先针对公司提出的需求进行了分析，并且确定了设计的大方向，笔者负责随后进行了总体设计，从系统组网、业务模型、配置指标等方面进行了规范，并且给出了总体设计理念。详细设计阶段，笔者依照总体设计的思想对模块进行设计，对内外接口和协议流程进行详细设计，并且组织编码人员完成实现。最后，还根据总体设计制定了测试方案，为全体开发人员和验证测试人员提供了具体测试依据和指导意见。

至此，基本完成了此课题设计，并且已经实现。无论从技术上说，还是商业角度上说，整个方案都是可行的，基本符合最初设想和需求。从总体上讲，此系统具有以下优势：

广泛支持各种协议转换

对协议簇进行了最大化设计，使其能支持短信，语音，UDP,HTTP 多种协议。

多模块结构和系统构建化

保证了信令系统能够根据要求网络现状进行灵活的容量和协议架构调整，满足不同容量、不同业务配置的要求。灵活满足客户不同组网、不同业务配置、不同容量各种信令网关的需求。

能力更加优化高效、系统更稳定

采用多语言设计，充分考虑了性能，稳定性，业务需求的变化。

在以上三大方面的优势和系统软硬件传统优势基础上，可以进行大规模的组网，实现功能。满足了用户的需求，为用户带来了可观的经济效益和社会效益。

6.2 未来展望

目前被系统已经应用于陕西省老年协会，在了解真正需求的时候，遇到了新的问题，目前正在积极的进行系统的新的设计，根据客户新的需求，要求腕式手表同时也可以进行与单个监护模块进行连接和数据传送，包括与后台的专家和医生管理平台连接；将成为这个阶段的主要工作，同时进行相关医疗设备入网认证。

把监护设备小型化化到手机里面， 进行连接和数据传送。

致谢

硕士论文付梓之际，研究生生涯行将结束。在这些个日日夜夜里，我付出了很多，也收获了很多，在这里我要感谢在这个期间帮助和关心过我的所有人，是你们为我的人生增添了一份精彩。

衷心感谢我敬爱的研究生导师——武波教授。在您的严格要求下，我对论文进行了很多次的修改；是您一次次评阅斧正我的论文，督促我将论文做到更好；是您一次次的教导，让我的治学态度更严谨；是您的言传身教，让我知道了该做一个怎样的人。是您不厌其烦的教诲，让我的论文逐渐丰满起来，从一个纯粹技术报告逐步变成一篇具有一定理论的论文，做到纲举目张，在这里，我还学到了很多文献编写的知识。

衷心感谢顾新教授百忙中抽出时间，对我论文提出了提供了无私的指导,尤其是一些细节的地方，例如用例，类图部分。感谢宋胜利老师对我数据库部分编写提出了指导；可以说我所取得的每一点成长和进步，都凝聚着老师们的心血。整个论文的编写过程，对我来讲，又是一个学习和实践的过程。

衷心感谢我的同学。感谢他们在生活中对我的帮助，感谢学校其他老师对论文的指导。有你们，我才会在这这么多日日夜夜里坚持下来，取得成就。

最后，衷心感谢在百忙之中评阅论文和参加答辩的各位专家、教授！

参考文献

- [1] 叶伟.互联网时代的软件革命 SaaS 架构设计.电子工业出版社,2009 :20-30
- [2] 华中宇 郝刚. silverlight2.0 入门指南.人民邮电出版社,2009 :5
- [3] 陈智鹏.走出山寨:MTK 芯片开发指南. 人民邮电出版社, 2010 :9
- [4] 唐学忠. UML 面向对象分析与建模. 电子工业出版社,2008 :11
- [5] 邱郁惠.系统分析师 UML 用例实战. 机械工业出版社,2010 :91-112
- [6] 刁成嘉. UML 系统建模与分析设计.机械工业出版社, 2007 :38
- [7] 郭宁. UML 及建模. 北京交通大学出版社,2006 :57-64
- [8] 谢星星,沈懿卓. UML 基础与 ROSE 建模实用教程. 清华大学,2008 :227-238
- [9] 刘晓华 等. UML 基础及 Visio 建模. 电子工业出版社, 2004
- [10] Philippe Kruchten. RUP 导论(原书第 3 版).机械工业出版社,2004 :35
- [11] (美)舒贾. IBM RUP 参考与认证指南.机械工业,2009 :7
- [12] (美)克罗列.敏捷与秩序/RUP 最佳实践.清华大学,2006.:22-29
- [13] 赵志新. MTK 手机开发入门. 人民邮电出版社,2010:21
- [14] 秦小波.设计模式之禅. 机械工业出版社. 2010 :58
- [15] 陈娴. ASP.NET 项目开发实践.中国铁道出版社,2003 :22
- [16] Craig Larman. UML 和模式应用. 机械工业出版社,2006
- [17] 王小科. C#开发实战宝典.清华大学出版社,2010:79
- [18] UML 数据库设计应用 naiburg.E.J 人民邮电出版社, 2002 :21-36
- [19] Cristian Darie. asp.net2.0 电子商务开发实战.人民邮电出版社, 2007:311-325
- [20] 梅尔斯.软件测试的艺术. 机械工业出版社,2006:7

附录 A

缩略语表

[illegible]