密级：秘密

文档编号：D0000-PPC-项目编号-PPD-年份

项目名称：**车辆管理系统**

项目编号：

车辆管理系统软件设计

东软集团股份有限公司 人才实训中心

(版权所有，翻版必究)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 总页数 |  | 正文 |  | 附录 |  | 生效日期 |  |
| 编制 | 闫瑞波 林金钏 孙高飞 郝雪 | | | 批准 |  | | |

文件修改控制

|  |  |  |  |
| --- | --- | --- | --- |
| **修改编号** | **版本** | **修改条款及内容** | **修改日期** |
| 1 | 1.0 | 创建 | 2014.7.25 |
| 2 | 1.0 | 修改 | 2014.7.29 |
| 3 | 1.0 | 完善 | 2014.8.8 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

[1 文档概述 4](#_Toc395253520)

[1.1 文档目的和范围 4](#_Toc395253521)

[1.2 术语/缩略语 4](#_Toc395253522)

[1.3 参考文档 5](#_Toc395253523)

[2 结构设计 5](#_Toc395253524)

[2.1 软件架构图 5](#_Toc395253525)

[2.2 模块说明 6](#_Toc395253526)

[技术点 6](#_Toc395253527)

[3 设计 7](#_Toc395253528)

[3.1 各模块详细类图 7](#_Toc395253529)

[3.1.1 业务人员： 7](#_Toc395253530)

[3.1.2 客服人员： 9](#_Toc395253531)

[3.1.3 审核人员： 11](#_Toc395253532)

[3.1.4 管理人员： 13](#_Toc395253533)

[3.1.5 服务器通信模块： 16](#_Toc395253534)

[3.1.6 数据库访问模块 17](#_Toc395253535)

[3.2 顺序图 18](#_Toc395253536)

[3.3 关键数据结构定义 19](#_Toc395253537)

[3.3.1 主要数据结构 19](#_Toc395253538)

[3.3.2 类成员函数名称 20](#_Toc395253539)

[4 数据库设计 20](#_Toc395253540)

[4.1 数据库引擎概述 20](#_Toc395253541)

[4.2 数据库概要设计 20](#_Toc395253542)

# 文档概述

## 文档目的和范围

**目的：**

从该阶段开始正式进入软件的实际开发阶段，本阶段完成系统的大致设计并明确系统的数据结构与软

件结构。主要是把软件需求转化为软件表示的过程，描绘出软件的总的概貌。

本文档为该系统的设计说明书，具体阐述了对用户所提出需求的设计[方案](http://www.7hao.net/index.php/Class/view/id-%B7%BD%B0%B8)，对系统中的各项功能需求、技术需求、实现环境及所[使用](http://www.7hao.net/index.php/Class/view/id-%CA%B9%D3%C3)的实现技术进行了明确定义，进一步细化软件设计阶段得出的软件总体概貌，把它加工成在程序细节上非常接近于源程序的软件表示。

**范围：**

本概要设计是基于需求分析的系统设计，适用于概要设计阶段、详细设计阶段、编码阶段、测试阶段

的项目指导。

## 术语/缩略语

|  |  |  |
| --- | --- | --- |
| 序号 | 术语/缩略语 | 说明 |
| 1 | API | 应用程序编程接口 |
| 2 | MFC | Microsoft Foundation Classes 微软基础类 |
| 3 | 保险人 | 要投保的人 |
| 4 | 客服人员 | 保险人可向客服人员报案，或预约投保 |
| 5 | 审核人员 | 审核保单，审核查勘记录并填写定损单 |
| 6 | 业务人员 | 给保险人投保，现场查勘，并提交审核 |

## 参考文档

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 文档名 | 作者 | 时间 | 版本 |
| 1 | 车辆保险理赔系统要求书.doc |  |  |  |
|  |  |  |  |  |

# 结构设计

## 软件架构图

业务员人员客户端

客户服务端

投保业务

理赔业务

预约投保

管理模块

审核业务

报案登记

控制器层

服务器

数据库访问层

数据库

图2-1 系统结构图

## 模块说明

|  |  |  |
| --- | --- | --- |
| 序号 | 模块 | 说明 |
| 1 | 审核业务 | 包括投保审核和理赔的审核 |
| 2 | 报案登记 | 保险人向客服人员报案，生成报案单 |
| 3 | 管理模块 | 管理人员和管理保险的种类 |
| 4 | 预约投保 | 保险人向客服人员预约投保，生成预约投保单 |
| 5 | 投保业务 | 业务人员确认保险人投保信息后，保险人填写保单，并提交审核 |
| 6 | 理赔业务 | 业务人员现场查勘，确认信息后 |
| 7 | 数据库访问层 | 利用DAO模式封装对数据的增删改查 |
| 8 | 控制层 | 将1,2,3,4,5,6模块封装好的数据按照一定的格式创建请求对象，通过控制层调用通信模块，与服务器交互 |
| 9 | 终端通信模块 | 与服务器建立socket通信，并进行数据交流 |
| 10 | 服务器 | 处理终端通信模块封装好发送过来的请求，调用服务器相应的模块，并返回结果给终端通信模块 |

### 技术点

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **技术点** | **技术点分类** | **技术点应用场景** |
| 1 | MFC基础控件使用 | List Control | 主界面及其他的展示界面。 |
| Edit Control |
| Button Control |
| Group-box Control |
| Text Control |
| Combo-box Control |
| 通过代码动态控制控件的显示、隐藏 |
| 2 | 线程 | 线程的创建、销毁 | 提醒功能的实现 |
| 3 | 数据库操作 | MySql增删改查 | 信息的保存 |
| 4 | Windows Socket编程 | 网络通信 | 终端通信模块与服务器的通信 |

# 设计

## 各模块详细类图

### 业务人员：

投保业务模块：



图3-1-1-1 业务员投保业务模块类图

理赔业务模块



图3-1-1-2 业务员理赔业务模块类图

### 客服人员：

报案登记



图3-1-2-1客服人员报案登记模块类图

投保预约



图3-1-2-2客服人员投保预约模块类图

### 审核人员：

审核预约投保



图3-1-3-1 审核人员审核预约投保模块类图

审核现场查勘



图3-1-3-2 审核人员审核现场查勘类图

### 管理人员：

管理员工信息



图3-1-4-1 管理人员管理员工信息类图

管理保险类型信息



图3-1-4-2 管理人员管理保险信息类图

### 服务器通信模块：



图3-1-5服务器通信模块类图

### 数据库访问模块



图3-1-6 数据库访问层类图

类图备注:图中实体类的get、set方法没有写，控制类中的主要的方法都写上了。

## 顺序图



图3-2-1 服务器通讯顺序图

描述：如图3-2-1所示为服务器通讯模块，首先Connctor类调用init和start方法进行初始化，接收客户端请求，当有一个客服端连接时，Connector中new一个task加入线程池中，在task中条用request解析客服端请求，处理完成之后调用response函数响应客户端的请求发送数据。



图3-2-2 客服人员添加预约投保顺序图

描述：如图3-2-1所示为客服人员添加预约投保顺序图，首先客服人员进入客服人员的主界面，点击左侧的树形操作命令中的“添加预约投保”条目，右侧区域进入添加预约投保界面，然后填写一系列响应的内容，点击提交按钮，在此过程中，BasicInsuranceController会实例化一个BasicInsurance实体类并填充响应的内容，Controller中的Socket将实体类以结构体的方式发送出去，服务器端Socket接收实体类的信息，调用BasicInsuranceDao中的addInsurance方法将实体类的信息保存到数据库中，同时返回一个是否添加成功的信息，显示在客户端主界面上。

## 关键数据结构定义

### 主要数据结构

**从上面的类图可以看出，主要的数据结构（实体类）有**：

BasicInsurance(预约投保信息), Report（报案单信息）, Warranty（保单信息），InsuranceType（险种信息）, LossAssessment（定损确认信息）, RecordAccidentInfo, RecordBasicInfo, RecordCarInfo, RecordInfo, DutyDecisionInfo（这五个是现场查勘信息）。

**服务器通信模块数据结构：**

struct data

{

char[15] dllName; //调用的模块的名字

char[20] method; //调用的模块内的方法名

int paramLen; //参数的长度

void\* param; //指向参数的指针

};

业务流程通用函数(每一个业务过程定义一个以下类型的函数)：

//第一个参数：终端传过来的数据，

//第二个参数：返回的数据的长度

//返回，要传到服务器的数据

typedef void\* (\*GENERAL\_PROC)(void\*,int \*);

### 类成员函数名称

实体类中的方法主要是一些get、set和操作符重载等方法。控制类中是一些操作实体的方法以及实体类的序列化和反序列化的调用。界面类中主要是一些响应的函数。

# 数据库设计

## 数据库引擎概述

由于MySQL支持远程操作数据库以及以下的一些优点，我们选择了MySQL数据库作为存储数据的仓库。

MySQL是一个关系型数据库管理系统，是最流行的关系型数据库管理系统，在应用方面MySQL是最好的RDBMS(Relational Database Management System：关系数据库管理系统)应用软件之一。MySQL是一种关联数据库管理系统，关联数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。MySQL所使用的SQL语言是用于访问数据库的最常用标准化语言。

## 数据库概要设计

数据表概述：

|  |  |  |
| --- | --- | --- |
| 序号 | 类名称 | 用途 |
| 1 | basic\_insurance | 预约投保单，记录基本的预约投保信息 |
| 2 | warranty | 保单信息 |
| 3 | insurace\_type | 保险的种类，包括名字和保险费用 |
| 4 | insurance\_work\_list | 跟踪某一个保单业务流程。包括业务人员，审核人员，状态等 |
| 5 | employee | 员工信息，包括部门，权限，密码等 |
| 6 | report | 报案单信息 |
| 7 | survey\_record | 查勘记录信息 |
| 8 | loss\_assessment | 定损单 |
| 9 | accident\_work\_list | 跟踪理赔业务的流程 |



图4-2-1 数据库设计图1



图4-2-2 数据库设计图2