**软件设计文档**

文档管理信息表

|  |  |
| --- | --- |
| **主题** | 项目数据流转系统软件开发设计文档 |
| **版本** | 1.0 |
| **内容** | - |
| **关键字** | - |
| **参考文档** | - |
| **创建时间** | 2013年4月13日 |
| **创建人** | 杨祎 |
| **最新发布日期** | - |

**文档变更纪录**

|  |  |  |
| --- | --- | --- |
| **更改人** | **日期** | **更改内容** |
| 杨祎 | 2013-4-13 | 创建文件 |
|  |  |  |
|  |  |  |
|  |  |  |

**文档主要评审意见**

产品组

|  |  |  |
| --- | --- | --- |
| **评审人员** | **日期** | **意见** |
| 章小明 | 2013-4-13 | - |

QA组

|  |  |  |
| --- | --- | --- |
| **评审人员** | **日期** | **意见** |
| 三维数据员 | - | - |

**目 录**

1 开发规划 1

1.1 开发人员 1

1.2 开发计划 2

1.3 开发环境和工具 3

1.4 开发规范 3

2 总体设计 4

2.1 概念术语描述 4

**2.1.1** 术语1 4

**2.1.2** 术语2 4

2.2 基本设计描述 5

**2.2.1** 系统总体逻辑结构图 5

**2.2.2** 系统部署结构图 6

2.3 主要界面流程描述 7

**2.3.1** 功能1界面流程 7

**2.3.2** 功能2界面流程 7

2.4 模块列表 8

3 数据结构 9

4 接口规范 10

4.1 <模块1 API> 10

**4.1.1** Interface1 10

**4.1.2** Interface2 10

4.2 <模块2 API> 11

4.3 <模块3 API> 11

4.4 <数据库API> 11

5 模块设计 12

5.1 Module1设计 12

**5.1.1** 模块设计描述 12

**5.1.2** 模块界面描述 12

5.2 Module2设计 13

6 附录 14

6.1 第三方组件 14

6.2 参考资料 15

6.3 附加文档 15

# 开发规划

## 开发人员

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **角 色** | **主要职责** | **负责模块** | **人员** | **备注** |
| 项目经理  PM | * 项目全面负责 * 项目设计 * 主要框架/模块编写 * 项目进度控制 | * 所有模块 | 杨祎 |  |
| 产品经理  PT | * 定义需求 * 产品监督 * 结果验证（测试） * 用户文档 | 无 | 杨祎 |  |
| GUI | * GUI设计 | * 所有模块 | 杨祎 |  |
| 程序员  DEV | * 编码 | * 所有模块 | 杨祎 |  |
| 测试 | * 测试 | * 所有模块 | 杨祎 |  |

## 开发计划

<附开发计划表>

## 开发环境和工具

开发工具

|  |  |
| --- | --- |
| **工具** | **作用** |
| QtCreator2.6.0 | 编码和界面设计 |
| QT | 数据流转系统软件功能运作开发 |
| Virsual Studio 2008编译器 | Windows系统编译器编译QT程序 |
| Window debug tool | 调式器 |

## 开发规范

|  |  |
| --- | --- |
| **文档名称** | **位置** |
| QT编码规范 |  |
| 系统目录规范 |  |
|  |  |

# **总体设计**

企业信息由大量数据构成，是企业业务过程中的组成部分。这些数据根据企业业务规则在组织之间流转形成数据流。通过企业经营过程所包括的生产（业务）过程，物资流，资金流都可以通过数据流这种方式表示，另一方面，企业中还有一部分事务处理完全以文档为中心的，例如企业的行政机构的日常办公（文档也是数据的一种呈现形式）。数据在组织之间的流转可以通过基于工作流的形式实现。

工作流中的数据：数据流转和业务过程密不可分。数据流是过程流的外在表现。从信息理解的角度，数据流研究数据的语法，而过程流研究数据的语义。

## 概念术语描述

### 数据流转

局域网间点对点数据接收与发送。

### 数据分发

原始作业数据由项目经理撑手，将项目数据分发至项目组长，项目组长再将作业数据分块分发至项目各小组长，各小组长再分发作业数据至各组下的作业员。作业员完成作业将数据分发给质量检测人员，质量检测人员检查发送过来的数据，检测该人员的数据，合格则汇总，并提交至项目经理审核；不合格，返回给作业员需要修改。

### 数据上传

质检员将合格的数据上传至服务器，并汇总作业完成情况，交予项目经理。

## 基本设计描述

<文字性的设计思路>

1. 发送数据

发送数组至指定一组用户。发送给谁，需要在用户（人员）管理项中管理人员。对当前在线人员分组，分组为：项目经理，项目组长，项目小组长，作业员，质检员。

1. 项目经理具有最高权限，可以选择发送给所有人，也可以选择发送给指定人和分组
2. 项目组长当前可对组长以下级别的人员分组和发送数据。
3. 项目小组长可对作业员分组管理，只对作业可发送数据。
4. 作业员可对质检量分组管理并发送数据。
5. 质检员可以作业员分组管理，接收到的数据通过质量检测，不合格则反馈给作业员，并通知其数据不合格理由。合格的数据将上传到服务器入库，并汇总给经理，汇报工作完成任务情况。
6. 不允许作业员与作业员之间数据的传输？同级级别的用户不有相互数据交互？如果能相互传数据，数据会变得无序，产生混乱的数据流转？

### 系统总体逻辑结构图

业务工作流由一系列事件之间的发生因果关系组成，过程描述如图1所示。

说明：方框表示事件，有向线段表示事件间的因果关系

图1 工作流过程描述

层/参与者

项目经理

项目组长

项目小组长n

项目小组长3

项目小组长2

项目小组长1

n作业员n~m

3作业员1~c

2作业员1~b

1作业员1~a

质检员

入库

图2 数据的过程流

作业分流

层/数据

项目作业数据

数据制作

数据制作完成

数据不合格

数据质量检查

合格入库

### 系统部署结构图

数据库服务器

内部客户机

内部客户机

内部客户机

内部客户机

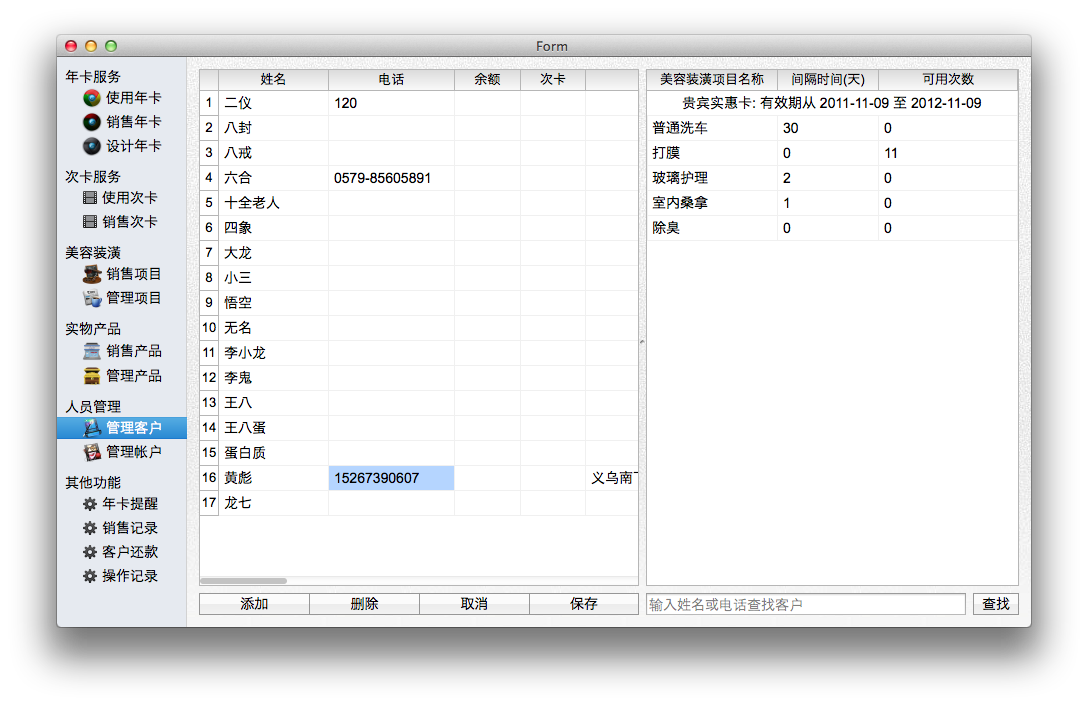
**系统网络拓扑结构**

INTERNET

## 主要界面流程描述

### 功能1界面流程

### 功能2界面流程



1

1. 经理
   1. 作业发放
   2. 历史明细
   3. 人员管理
   4. 其他功能
   5. 设置
2. 组长
   1. 作业接收
   2. 作业发放
   3. 历史明细
   4. 小组管理
   5. 其他功能
   6. 设置
3. 作业员
   1. 作业接收
   2. 历史明细
   3. 质检管理
   4. 其他功能
   5. 设置
4. 质检员
   1. 作业接收
   2. 作业反馈
   3. 作业入库
   4. 作业员管理
   5. 其他功能
   6. 设置

## 模块列表

|  |  |  |
| --- | --- | --- |
| **模块名称(英文)** | **功能** | **备注** |
| 模块1  Module1 |  |  |
| 模块2  Module2 |  |  |
| 模块3  Module3 |  |  |

# 数据结构

数据库表列表

|  |  |
| --- | --- |
| **表名** | **作用** |
|  |  |

<附SQL数据库结构定义文档>

# 接口规范

## <模块1 API>

* **描述**

<模块功能和设计描述(简要描述)>

用户类：基类User(用户)

继承类：经理用户，作业员组长用户，作业员小组长用户，普通作业员用户，质量检查用户

用户操作类：数据接收和发送

用户权限类：

* **集成和部署**

<描述模块如何同系统集成，实现形式是DLL, EXE或是嵌入式代码>

### Interface1

<interface1 description>

* Function1

**string function1(int param1, string param2, …)**

<function1 description>

param1 - <param1 description>

param2 - <param2 description>

* Function2

**string function2(int param1, string param2, …)**

<function2 description>

param1 - <param1 description>

param2 - <param2 description>

### Interface2

## <模块2 API>

## <模块3 API>

## <数据库API>

定义数据库存储过程接口

# 模块设计

## Module1设计

<根据具体情况细化模块设计>

### 模块设计描述

使用QTcpServer组件作为侦听的服务器接收数据

使用QTcpSocket组件作为连接服务器和发送数据

1. QTcpServer的服务器地址与端口需要在网络中广播出去,让其他TcpSocket客户端能连接上，所以Server端需要一个UdpSocket来广播自己的服务器ip和port。
2. TcpSocket通过以接收到的广播地址和端口，来connectToHost连接上去。将连接的主机加入到列表。
3. 以后要与该主机进行通信发送数据，则在线程中进行连接并收发数据。

sendBroadcastData

TcpServer

TcpSocket

Broadcast

Bind

Connect

Listen

Receive connect

readBroadcastData

Recieve

Receive

Send

Close

Send

Recieve

Close

图5 局域网间数据收发流程结构图

### 模块界面描述

## Module2设计

# 附录

## 第三方组件

QT组件

|  |  |  |
| --- | --- | --- |
| **组件名称** | **版本** | **用途** |
| QtCore4 | 4.8.0 | QT程序核心接口 |
| QtGui4 | 4.8.0 | 软件界面GUI接口 |
| QtNetWork4 | 4.8.0 | 软件网络模块接口 |
| QtSqLite | 4.8.0 | 软件数据库模块接口 |

.NET组件

|  |  |  |
| --- | --- | --- |
| **组件名称** | **版本** | **用途** |
| MS .NET | 3.5 |  |

## 参考资料

Design Pattern -

## 附加文档

|  |  |  |
| --- | --- | --- |
| **文档名称** | **文件名** | **描述** |
| 项目计划表 | Wind.net.plan.mpp | 项目时间规划 |
| 数据库定义 | X\*\*\*\*.sql | 数据库定义脚本 |
| \*\*\* |  |  |
|  |  |  |
|  |  |  |