通用审批管理软件V1.0

设计说明书

# 1引言

## 1.1编写目的

《通用审批管理软件》的目的在于从设计的角度明确软件的功能范围，帮助用户了解软件，更好更快地应用软件，同时方便软件维护人员维护软件、升级软件。该说明书也将作为开发者前期开发的重要依据。开发者希望通过该说明书能让开发者和用户更好地沟通，并作为后期交付时软件是否合乎用户标准的一个重要参考。

## 1.2背景

审批是一个企业日常办公过程中必不可缺的一项内容。不同工作领域都有各自不同的审批需求和方式。在企业管理当中，每天都会有大量的申请信息需要上级审批，由于人员分散、流程复杂等原因，往往耗费大量时间，降低办公效率。随着电子化办公的日益普及，企业迫切需要一个基于网络的规范审批流程。我们的通用审批管理软件正是这一背景下软件体系结构课程的课程作业，本组成员针对多领域的审批工作需求、复杂繁琐的审批流程、多样的审批角色和用户特点[1]，提供了专业的审批管理系统，以简化管理流程、提高审批效率。

## 1.3定义

（1）流程模板：管理员为满足特定行业审批需求，通过服务器端构建的一个规范的审批流程。

（2）原型模型：指在执行实际软件的开发之前，应当建立的一个系统工作原型，在该原型的基础上，逐渐完成整个系统的开发工作。

（3）流程模板/模板：管理员在服务器端定义的事务处理流程逻辑结构。

（4）单支节点：模板中只有一个用户参与审核的步骤。

（5）分支节点：模板中有多个用户共同参与审核的步骤。

## 1.4参考资料

[1] 裘虹.基于J2EE框架的通用审批管理系统.万方数据.2012

# 2开发计划

## 2.1项目概述

### 2.1.1工作内容

本项目采用C/S架构，分别做了客户端和服务器端的程序。

服务器端主要工作：

（1）构建流程模板

（2）接收并存储用户身份信息

（3）接收并存储用户申请

（4）控制申请项进行状态

（5）向客户端返回用户申请和待处理信息

客户端主要工作：

（1）向用户提供交互界面

（2）向用户提供注册、登录、申请、审批等功能

（3）向服务器提交用户操作信息

### 2.1.2主要参加人员

## 2.2实施计划

### 2.2.1工作任务的分门与人员分工

本项目的任务分门如下：

（1）构建模板：设计每个流程结点的结构以及流程模板的存储方式。

（2）网络通信：解决服务器如何响应多个客户端的连接请求的问题。

（3）信息解析：设计客户端与服务器之间发送信息的格式，以进行相应操作。

（4）流程控制：在服务器端对用户的申请和审批操作进行响应。

具体的人员分工如下。

（1）张晓倩：统筹全组工作、系统原型模型的构建、信息解析设计；

（2）方子若：流程模板存储设计、流程进行控制模块；

（3）罗佳喻：网络通信设计、软件测试；

（4）张煜曼：流程模板结构图形化显示、PPT制作；

### 2.2.2进度

第一周：收集用户需求，并构建系统原模型；

第二周：划分任务，分工合作；

第三周：代码整合，功能细化；

第四周：撰写软件设计说明书和制作汇报PPT。

### 2.2.3关键问题

设计一个好的模板和用户申请的存储形式，保证能够按照构建好的模板正确地处理用户的申请或审批操作。

# 3需求分析

## 3.1任务概述

### 3.1.1目标

本软件是由于《软件设计与体系结构》这门课程产生，适用于中小型企业，目的是改善目前很多企业内部由于人员分散、申请流程复杂而造成的低办公效率的现状。在此应向读者说明的是，本软件产品是一项独立的软件，而且全部内容自含。

### 3.1.2用户的特点

用户特点如表3-1所示。

表 3‑1用户特点

|  |  |
| --- | --- |
| 用户类 | 特征 |
| 维护人员 | 专业的软件编程人员，熟悉网络通信服务，并能够熟练运用VB.NET编程语言。 |
| 管理人员 | 企业管理人员，负责审批流程的管理，较熟悉电脑操作。 |
| 操作人员 | 企业普通员工或管理人员，有使用软件进行申请或审批的需求，懂得基本的电脑操作。 |

### 3.1.3运行环境

（1）硬件环境

系统的硬件环境如表3-2所示。

表格 3‑2 硬件环境

|  |  |
| --- | --- |
| CPU | 1G Hz以上 |
| 内存 | 1G以上 |
| 硬盘 | 10G以上 |

（2）软件环境

系统的软件环境如表3-3所示。

表 3‑3 软件环境

|  |  |
| --- | --- |
| 操作系统 | Windows 10 64位 |
| .NET Framework | 4.0 |

## 3.2功能性需求

### 3.2.1构建模板

表 3‑4 构建模板

|  |  |
| --- | --- |
| 功能描述 | 管理员根据需求构建申请审批流程模板，提供分支选择功能 |
| 输入 | 1. 用户类型 2. 模板结点信息，包括：响应的用户类型、操作类型、操作约束等 |
| 加工 | 1. 添加用户身份，同时对输入进行有效性检查； 2. 构造流程模板中的节点项；   E.2.1若要增加节点，则选择用户，输入操作类型、操作约束，点击添加节点按钮；  E.2.2若要删除模板中的节点，选择列表中节点项，点击删除节点；  3. 构建模板部分结束。 |
| 输出 | 一个实例化的流程模板 |
| DFD图 |  |
| 注释 | 操作约束如果有的话，形式必须固定 |

### 3.2.2根据模板显示流程图

表 3‑5 根据模板显示流程图

|  |  |
| --- | --- |
| 功能描述 | 将实例化的流程模板可视化显示 |
| 输入 | 实例化的流程模板 |
| 加工 | 1. 对流程节点中包含的数据进行有效性检查 2. 根据实例化流程模板构建开始、结束矩形和流程中的节点集合； 3. 根据节点集合和设计好的表现形式在窗口中画出流程图。 4. 显示流程图部分结束 |
| 输出 | 审批流程的图形化表示 |
| 流程图 |  |
| 注释 | 无 |

### 3.2.3处理申请

表 3-6 处理申请

|  |  |
| --- | --- |
| 功能描述 | 响应申请者的申请操作 |
| 输入 | 申请者信息、申请内容的数目 |
| 加工 | 1. 将申请者的学号和申请数目发送给服务器； 2. 服务器接收申请内容，并加入到申请列表； 3. 服务器根据流程模板对查找该申请的审批人，并写入申请列表。 |
| 输出 | 一个新的申请项 |
| DFD图 |  |
| 注释 | 无 |

### 3.2.4 处理审批

表 3‑7 处理审批

|  |  |
| --- | --- |
| 功能描述 | 响应审批者的审批操作 |
| 输入 | 待审批申请项的用户ID、申请数目、审批者身份 |
| 加工 | 1. 服务器接收审批内容； 2. 根据申请者ID和申请数目，定位到一个申请项； 3. 将该申请项内待审批人中该审批者身份删除； 4. 若该申请项的待审批人部分为空，转步骤5，否则转步骤6； 5. 根据流程模板添加下一层中应参与审批的审批者身份； 6. 审批状态更新完成。 |
| 输出 | 审批状态更新过的申请项 |
| DFD图 |  |
| 注释 | 无 |

## 3.3非功能性需求

### 3.3.1性能需求

（1）时间特性要求

同时使用本软件的用户数量小于100时，系统能够在10s内完成客户端和服务器间的数据传输，并在5s内正确地响应用户的操作。

（2）灵活性

本软件的运行环境为.NET Framework 4.0及以上版本。并且给予管理人员在服务器端自定义审批流程的权限，如遇企业内部制度改革，管理人员可轻易地修改流程模板，完成系统的更新换代。

### 3.3.2安全性需求

（1）系统保密性：申请项信息只有客户端的申请者和服务器端的管理员能够查看；

（2）系统完整性：申请者能够看到完整的申请信息，包括申请的审批进度；

（3）可用性和抗毁性：设置备份机制，关闭服务器窗口时，将所有用户信息和申请信息写入到磁盘文件中。

### 3.3.3软件质量属性

（1）正确性：系统在预定坏境下能正确地响应用户操作；

（2）健壮性：在用户输入的数据无效或操作错误等意外情况下，系统能够做出适当的响应消息；

（3）可测试性：测试人员通过同时开启多个客户端且进行多项操作对软件进行测试。

（4）效率：系统运行过程中，约占用客户端10M的内存资源；

（5）可理解性：用户参照用户手册可轻易掌握软件的使用方法和注意事项；

# 4概要设计

## 4.1总体设计

### 4.1.1处理流程

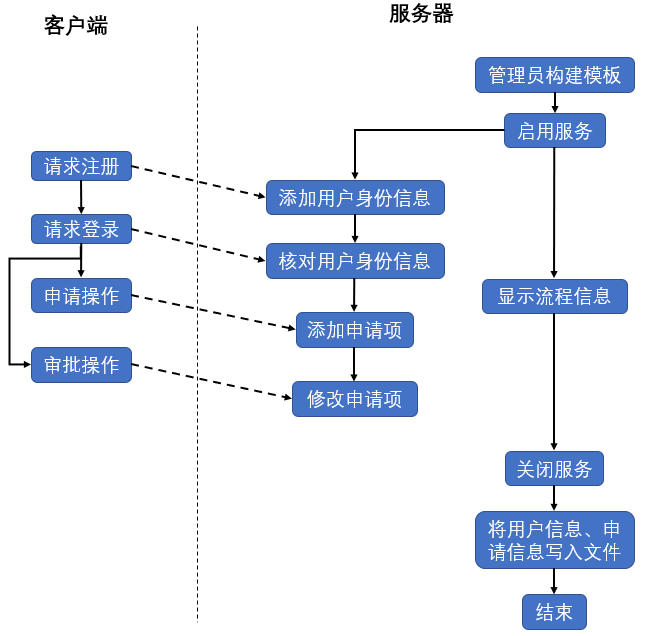


图 4-1 处理流程

### 4.1.2系统模块结构图

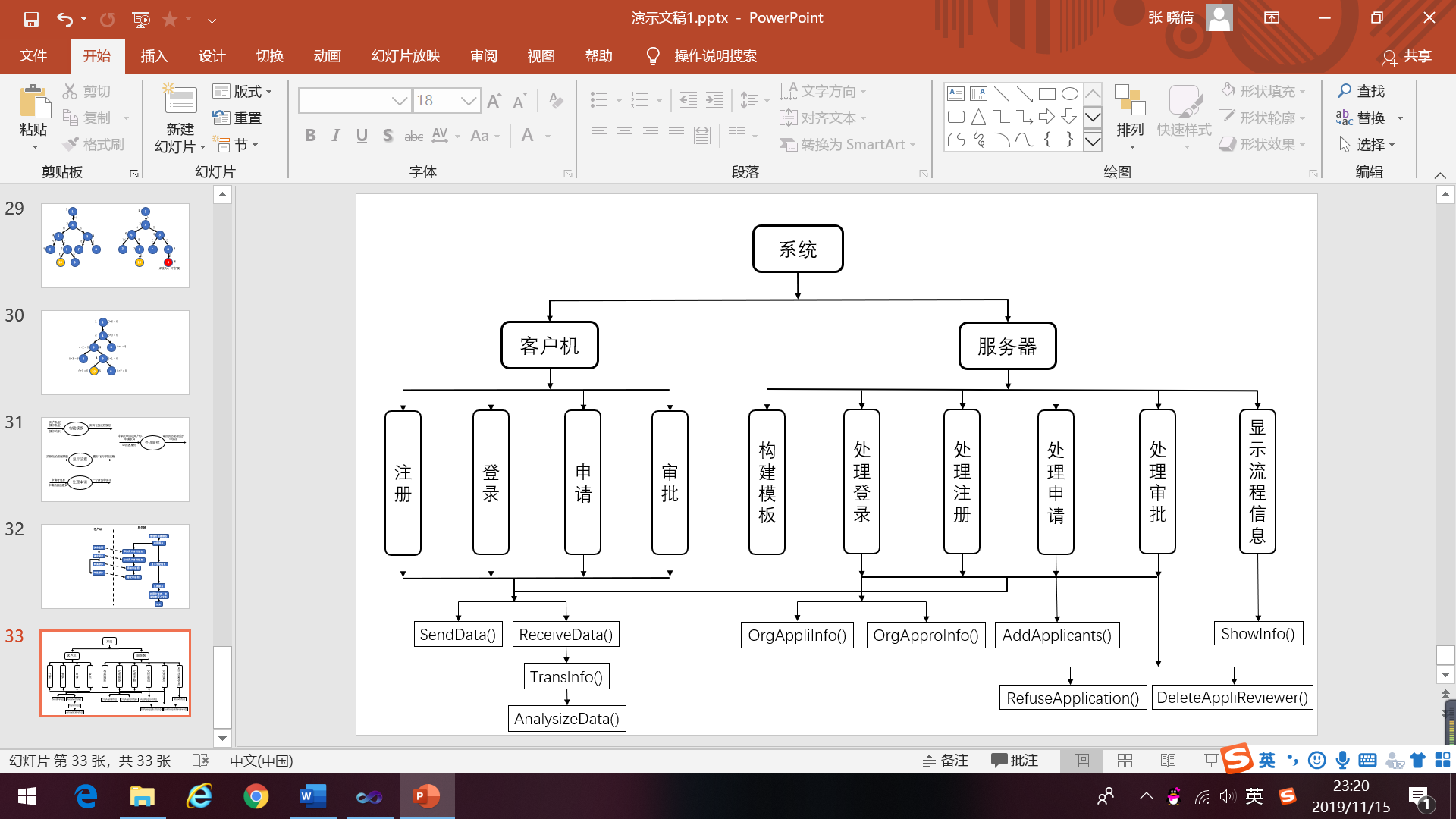


图 4-2 系统模块结构图

系统模块结构图如图4-2所示。

客户机程序分成注册、登录、申请、审批四个模块，每个模块都需要向服务器发送信息和接收来自服务器的信息。

（1）注册模块：在系统中注册用户；

（2）登录模块：用户登录，查看相关信息，进行申请或审批操作；

（3）申请模块：在用户信息页面，具有申请者身份的用户进行申请操作；

（4）审批模块：在用户信息页面，具有审批者身份的用户选择一个审批项进行审批操作，可以选择“同意”或“不同意”。

服务器程序分成：构建模板、处理注册、处理登录、处理申请、处理审批、显示流程信息五个模块。每个模块都需要向客户机发送信息和接收来自客户机的信息。

（1）构建模板模块：系统投入使用前，管理员通过服务器的否构建模板功能构建一个审批流程的模板；

（2）处理注册模块：接收客户机发送的用户注册信息，并添加至用户信息数组；

（3）处理登录模块：接收客户机发送的用户登录信息，整理用户相关的申请或审批信息并发送给客户机

（4）处理申请模块：接收客户机发送的申请信息，新建申请项并加入申请项列表；

（5）处理审批模块：接收客户机发送的审批信息，对申请项列表中相应申请项信息进行更新；

（6）显示流程信息：在系统运行过程中查看所有的申请项信息。

标识符及功能说明如表4-1所示。

表 4-1 标识符及功能说明

|  |  |
| --- | --- |
| 标识符 | 功能 |
| SendData() | 向客户机或服务器发送信息 |
| ReceiveData() | 接收客户机或服务器发送的信息 |
| TransInfo() | 去掉接收信息的byte字节数组中的无效字符，保证信息长度的准确性 |
| AnalysizeData() | 对接收到并经过长度处理的信息根据设计好的协议进行分隔 |
| OrgAppliInfo() | 服务器端根据接收到的申请者登录信息，组织该申请者的所有申请项信息 |
| OrgApproInfo() | 服务器端根据接收到的审批者登录信息，组织该审批者待审批的所有申请项信息 |
| AddApplicants() | 服务器端新建申请项并加入申请项数组 |
| RefuseApplication() | 服务器端处理审批者对申请项的拒绝操作 |
| DeleteAppliReviewer() | 服务器端处理审批者对申请项的同意操作 |
| ShowInfo() | 服务器端管理员查看所有申请项信息 |

### 4.1.3尚未解决的问题

服务器端无法为每一个请求连接的客户机都创建一个线程，因为这涉及到临界区的管理。

## 4.2用户接口设计

如图4-3所示，管理员在服务器端可以进行构建模板、其用户服务、关闭服务、显示流程信息等操作。



图 4-3 服务器端主页

如图4-4所示，管理员进入构建模板页面后，可以添加或删除用户类型和流程节点、显示构件好的流程图。

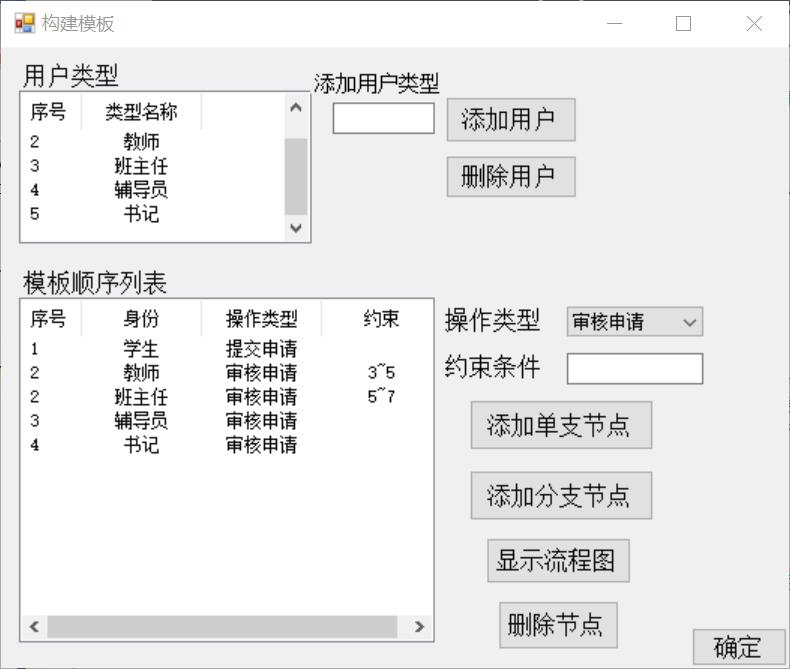


图 4-4 构建模板界面

如图4-5所示，在客户机端首页，用户输入自己的ID、密码、身份进行注册或登录，如图 所示。



图 4-5 客户端首页

如图4-6所示，登录进入主页后，申请者点击提交申请按钮，在弹出的申请页面中输入申请数目，按确定键完成申请操作。

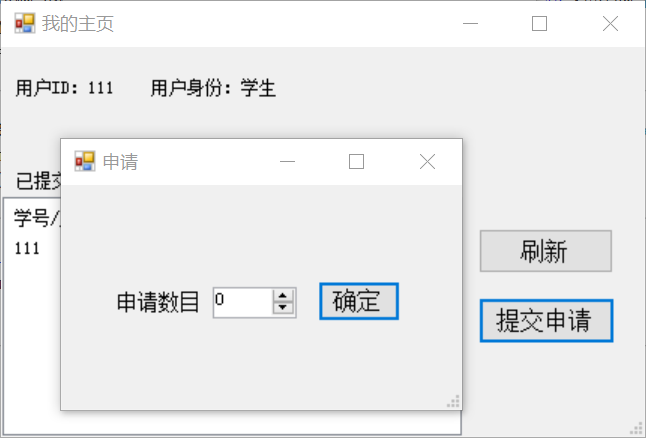


图 4-6 客户端用户主页进行申请操作

如图4-7所示，审批者选中自己待审批的某一项后，点击审核申请按钮后，在弹出的审批页面中选择“同意”或者“不同意”申请。

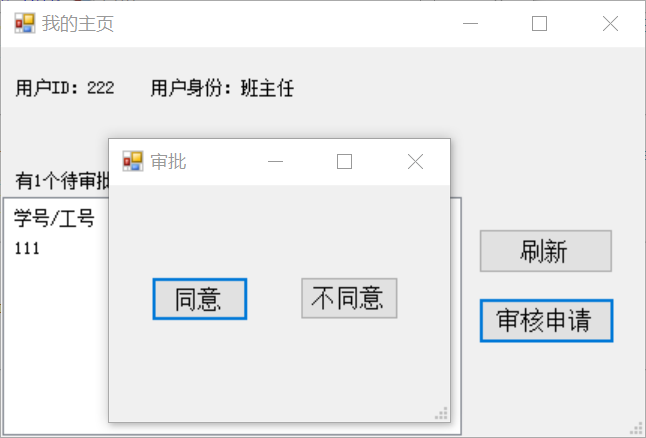


图 4-7 客户端用户主页进行审批操作

## 4.3系统出错处理设计

### 4.3.1出错信息

用户提交申请或审批信息后，刷新页面，可能发现系统未响应之前的操作。

### 4.3.2补救措施

重启客户端软件，再次进行操作。

## 5详细设计

## 5.1添加流程节点

### 5.1.1程序描述

将用户选择、由响应者身份、约束条件、操作类型、节点类型构成的流程节点加入到流程模板数组中。

### 5.1.2程序设计

steps是Eachstep类型的列表。

（1）由用户身份、操作类型、约束条件创建并初始化一个Eachstepitem类型的对象，命名为eachstepitem；

（2）获取节点类型，若为单支节点，继续，否则转步骤（6）；

（3）新建Eachstep类型的对象，命名为eachstep；

（4）eachstep.eachsteplist.Add(eachstepitem)；

（5）steps.Add(eachstep)，结束；

（6）将eachstepitem追加到steps[steps.count - 1]的eachsteplist列表中，结束。

### 5.1.3流程逻辑

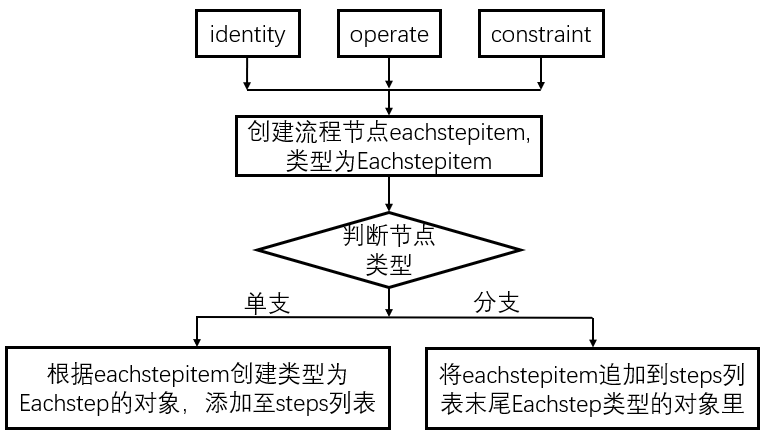


图 5-1 添加流程节点流程逻辑

## 5.2显示流程图

### 5.2.1程序描述

将构建好的流程模板图形化显示为流程图。

### 5.2.2程序设计

（1）根据窗口大小，计算流程图中各类结点的大小、位置；

（2）创建clsRectangle类对象，该类包含矩形画法，绘制“开始”结点；

（3）新建clsEllipse类对象集合，该类包含椭圆的画法以及椭圆内文字内容的绘制，绘制所有流程内容节点；

（4）新建clsRectangle类对象，绘制“结束”节点；

（5）建立clsLine类，创建对象CLine对所有图形进行连线：

a）若后续节点是单支节点，将上一行所有节点与后续节点相连；

b）若后续节点是分支节点，新建clsDiamond类分叉，使上一行所有节点经过分叉与后续节点相连；

（6）若窗口大小发生改变，转步骤（1）。

### 5.2.3流程逻辑

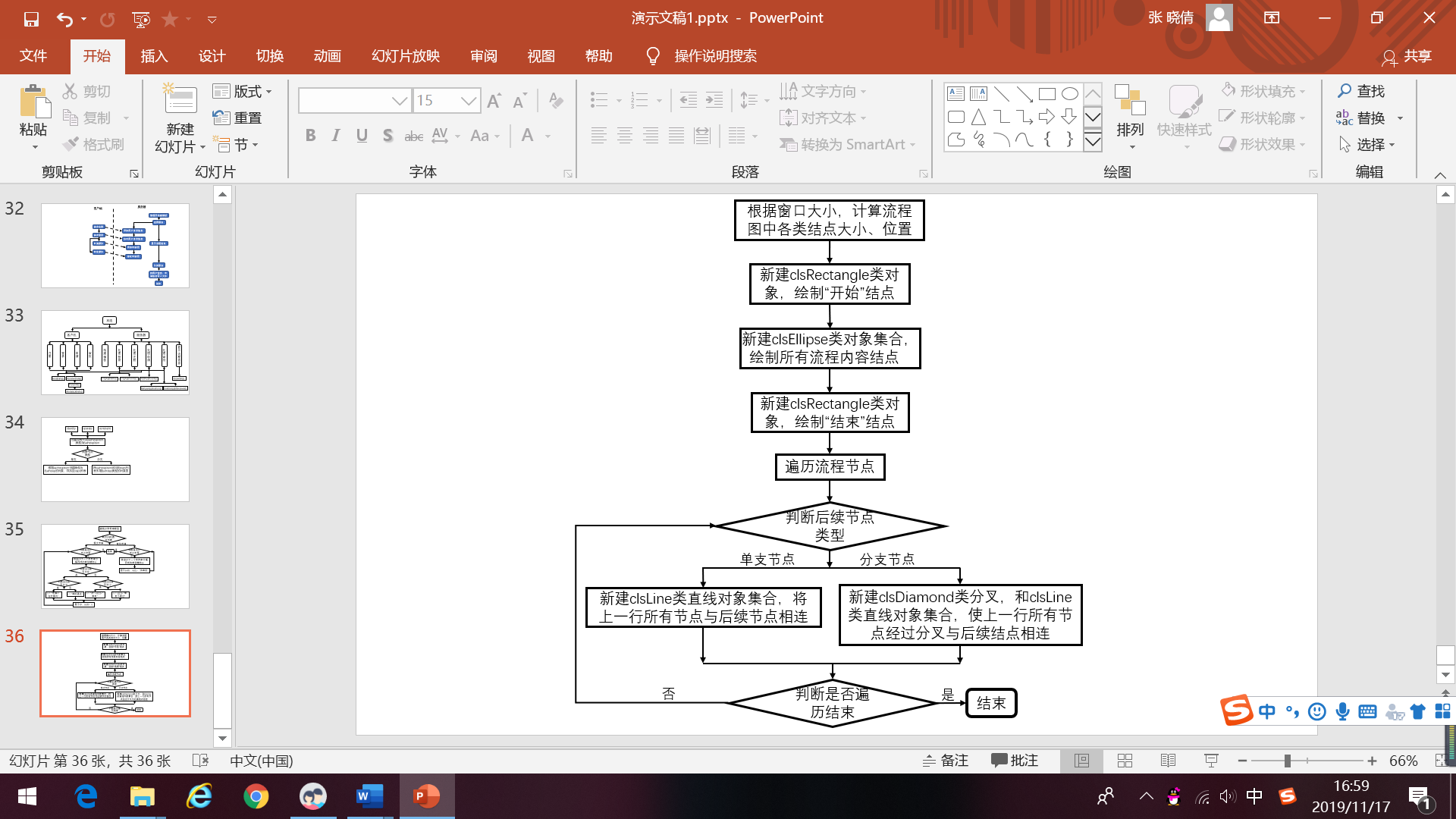


图 5-2 显示流程图流程逻辑

## 5.3解析登录时接收信息

### 5.3.1程序描述

用户登录后，会接收到服务器传来的用户申请或待审批信息，客户机程序需要对这些信息进行解析，以便显示。

### 5.3.2程序设计

接收到的信息已经过分隔，并存在str字符串数组中。

（1）遍历str数组，若str[0]值为“提交申请”，转步骤（2），否则转步骤（8）；

（2）若str数组已至末尾，结束，否则取str数组下一个字符串；

（3）将取出字符串分隔而成的数组赋给ss；

（4）若ss[1]不为空，继续，否则转步骤（6）；

（5）若ss[2]为”0”，将申请信息项s解析为“申请不在范围内”，否则解析为“审批通过” ；

（6）若ss[3]为[1]，将申请信息项s解析为“待”+ss[1]+“审批”，否则解析为ss[1]+“审批不通过”；

（7）显示申请信息项：ID、ss[0]、s，转步骤（2）；

（8）将取出字符串分隔而成的数组赋给ss；

（9）显示待审批信息项：ss[0]、ss[1]、“待审批”，转步骤（2）。

### 5.3.3流程逻辑

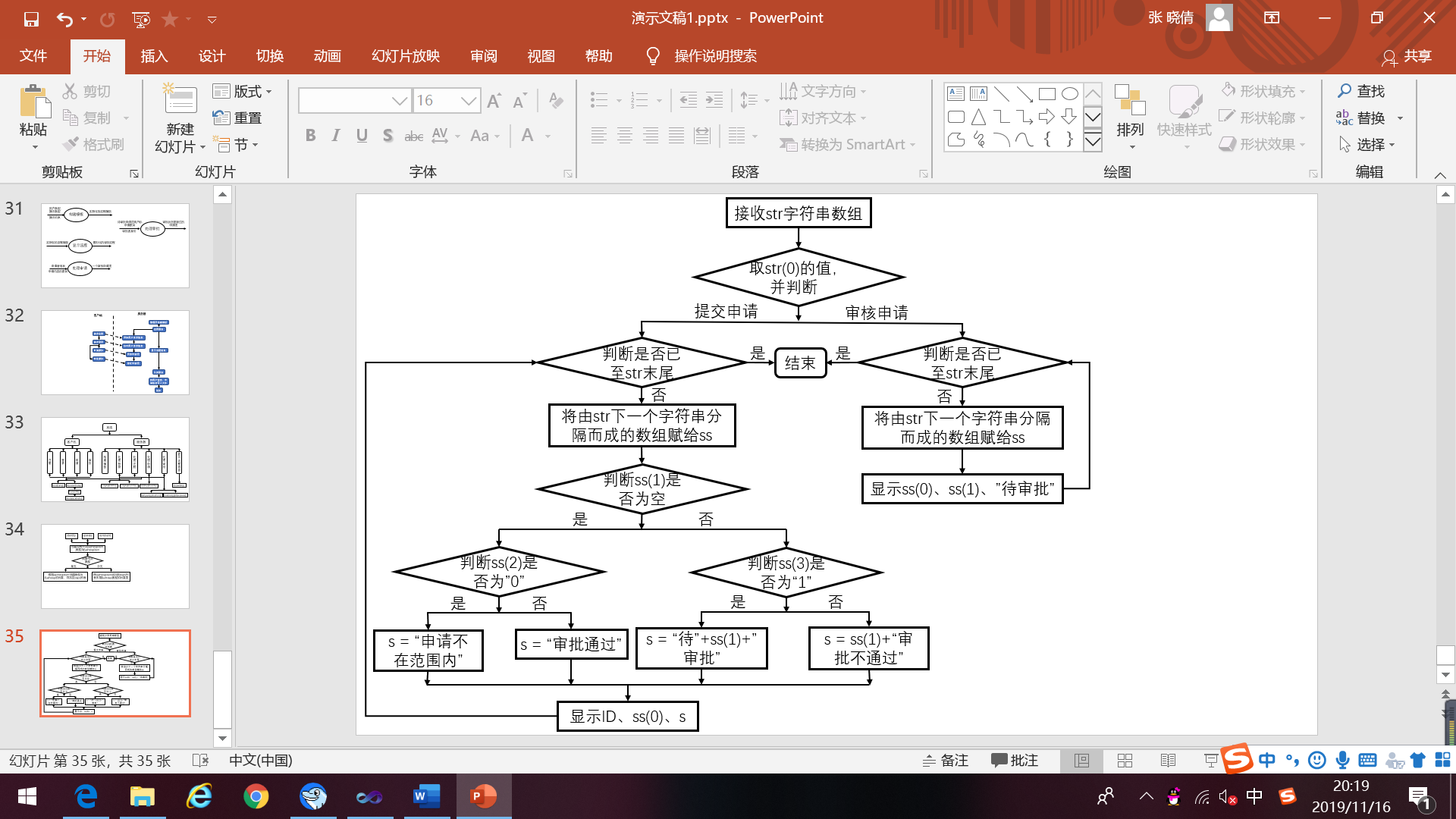


图 5-3 解析登录时接收信息流程逻辑

# 6项目总结

## 6.1开发成果

### 6.1.1主要功能和性能

表 6-1 主要功能和性能

|  |  |
| --- | --- |
| 功能 | 完成情况 |
| 构建模板 | 完全达到预期，但交互不够便捷。 |
| 注册 | 完全达到预期。 |
| 登录 | 完全达到预期。 |
| 申请 | 基本达到预期，但多个客户端同时运行会出现申请项被吞掉的情况。 |
| 审批 | 基本达到预期，但多个客户端同时运行也会出现审批操作被吞掉的情况。 |
| 显示所有申请流程信息 | 完全达到预期。 |
| 存储所有申请信息 | 未达到预期，本想使服务器连接数据库，但最终只搞了文本文件存放最终数据。 |

### 6.1.2性能

（1）响应时间：不出错情况下，系统能够在1s内响应用户操作。

（2）并发用户数：能够支持10名客户端同时使用。

### 6.1.3进度

项目进度如下，基本符合开发计划：

第一周：收集用户需求，并构建系统原模型；

第二周：划分任务，分工合作；

第三周：代码整合，功能细化；

第四、五周：撰写软件设计说明书和制作汇报PPT。

## 6.2开发工作评价

### 6.2.1生产效率

对技术的掌握与应用程序较高。

### 6.2.2产品质量

服务器端构建模板时需要用户仔细阅读操作手册进行操作，不然客户端运行时会出错。

### 6.2.3技术方法

构建流程模板时，我们理解到每个流程节点其实都规定了哪种类型的用户需要对申请事件进行响应（即某种操作）。于是，我们将流程节点映射为一个个包含响应用户类型、用户操作类型、节点类型（分支或单支），那么整个流程就可以用一个二维数组来表示，如图6-1所示。如此，对流程节点的创建和删除就可以抽象为对数组元素的添加和删除。

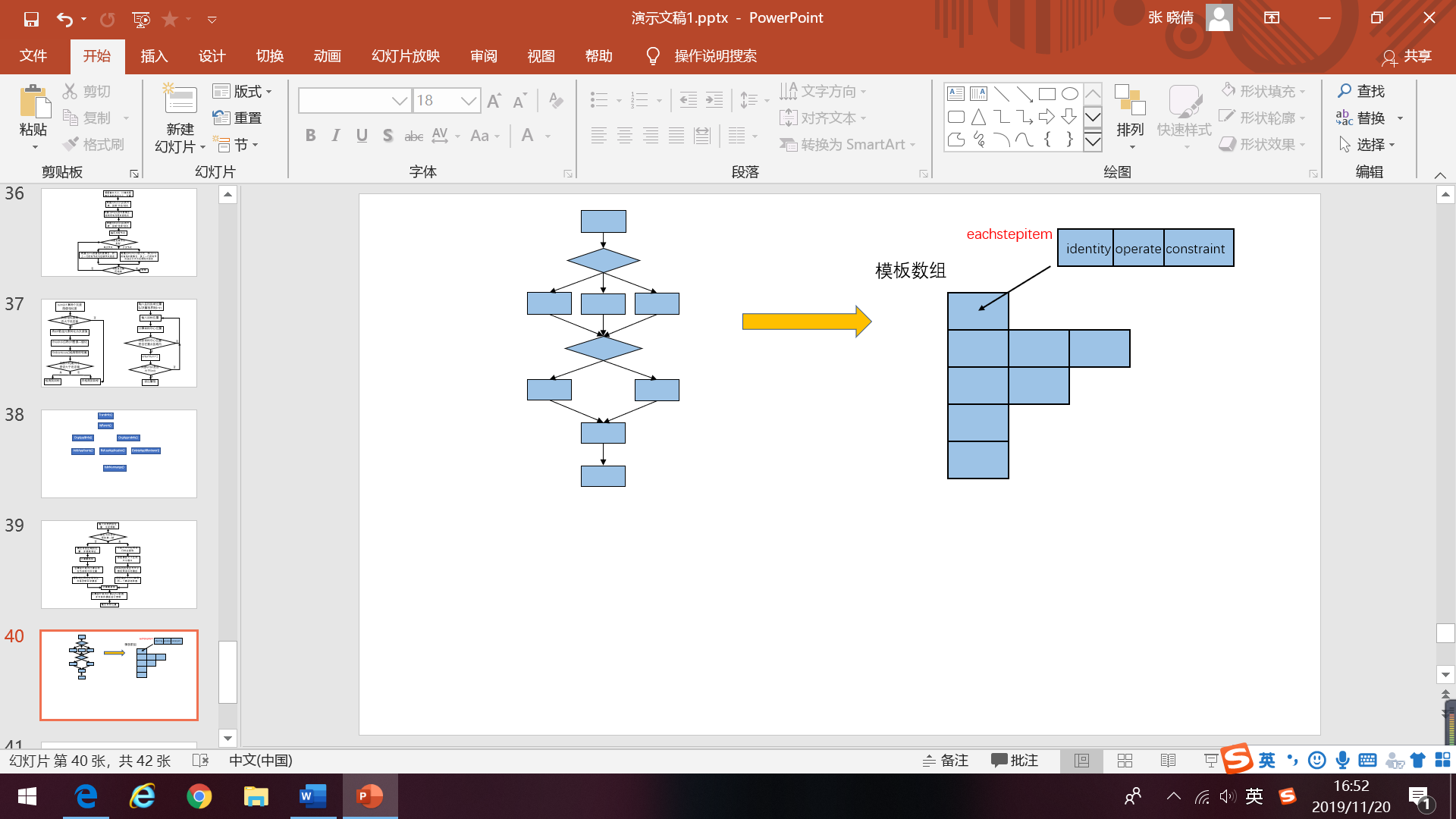


图 6-1 存储流程模板设计方法

### 6.2.4出错原因

未实例化问题：

（1）必须先在服务器端构建好一个流程模板，才能运行客户端程序。

（2）流程中有且只有一个提交申请的节点，且必须为初始节点。

## 6.3经验与教训

这次作业是入学以来除了学期初的课设外第一个要互相合作、定需求、做概要设计、编程序的任务。编写该通用审批管理软件历时一个多月，其中核心程序开发时间两个星期，期间经过多次更正与优化，我们小组的最终成品实现了C/S系统结构，能够按照用户需求对模板结构进行设计，供登录用户进行申请和审批处理。

从最初逻辑结构设计到最终实现成品，一路走来，我们收获丰硕。首先，小组成员自行设计了网络传输协议，并以此为基础编写出C/S结构的系统。其次，更加清楚的认识到在软件开发过程中，合理规划的重要性，我们小组在讨论出大致雏形后就开始分工编写，并没有形成接口共识，导致任务的耦合度较高，后续合并代码和增加功能产生了不少麻烦。最后，本次项目锻炼了我们设计软件体系结构，合作完成任务和编写软件设计说明书的能力，也给我们长了记性：一定要先做好需求分析和详细设计再进行编码！这一点非常重要！

# 7 用户手册

## 7.1 系统简介

本通用审批管理系统为C/S结构，分为客户端和服务器两部分。客户端供用户注册、登录、进行事务处理，服务器端为客户端提供数据支持与初始化流程的构建。

## 7.2 系统流程介绍

（1）系统刚启用时，需先服务器端构建用户模板确定流程逻辑与不同身份用户权限，然后可选择开启事务流程处理服务。

（2）在服务器开启事务处理功能后，用户可以不同身份注册，注册成功，即可登录系统，提出申请，或事务处理。

（3）服务器关闭流程服务功能后，客户端将无法进行注册、登录、事务处理等操作。

## 7.3 操作说明

### 7.3.1服务器端

（1）服务器运行后，首页界面如图7-1所示。



图 7-1 服务器首页

（2）在主页选择“构建模板”按钮，进入构建模板界面，如图7-2所示。

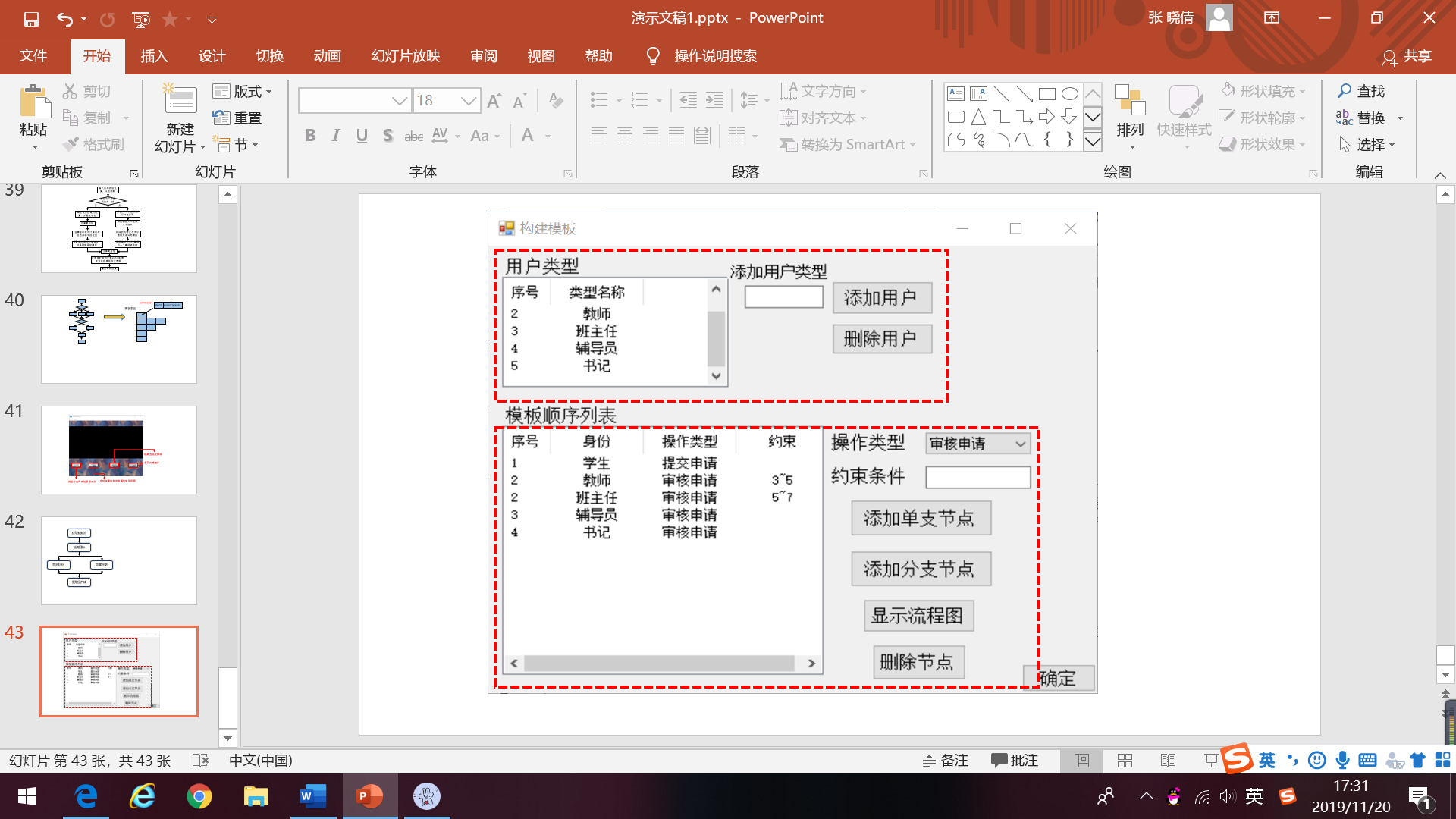


图 7-2 构建模板页面

构建模板功能可以按照客户的需求，搭建符合自己所需业务逻辑的工作流程模板，搭建完成后，此模板可以投入运行，供模板中设定的业务人员进行业务处理。

各类操作使用方法如下：

添加用户类型：先在用户输入框里输入用户类型，然后点击“添加用户”，即可添加；

删除用户类型：在用户类型列表框中选中一列，然后点击“删除用户”，即可删除该用户类型；

添加单支节点：在用户类型列表框中选中一列，选择“操作类型”，点击“添加单支节点”，即可添加单支节点；

添加分支节点：在用户类型列表框中选中一列，选择“操作类型”，点击“添加分支节点”，即可添加分支节点；

删除节点：在模板顺序表中选中一列，选择“删除节点”，即可；

显示流程图：在构建模板的过程中，可点击“显示流程图”查看图形化流程图，以供参考，如图7-3所示；

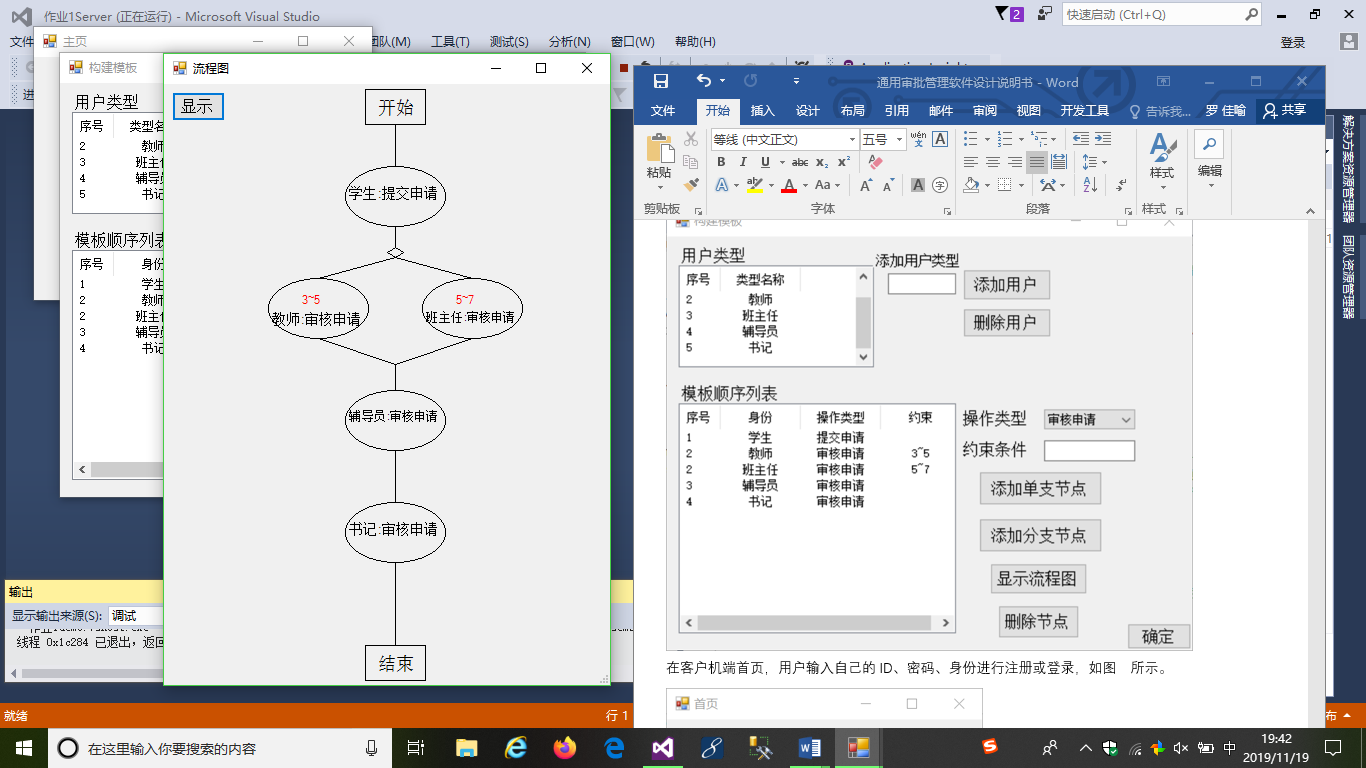


图 7-3 显示流程图界面

保存流程模板：点击“确定”。

### 7.3.2客户端

（1）客户端运行后，首页界面如图7-4所示。



图 7-4 客户端首页

（2）用户若还未注册账号，则输入ID、密码、身份进行注册，若该账号已注册过，系统会给出提示。

（3）用户注册过账号后，输入ID、密码、身份进行登录，若该账号还未注册过，系统会给出提示。

（4）用户登录后，进入登录页面。页面显示用户ID、身份。

如图7-5所示，若登录用户为申请者身份，则页面还会给出已提交申请项数，列表框内显示已申请项的申请数目和审批状态。点击“提交申请”按钮，用户可以选择申请数目并进行申请。

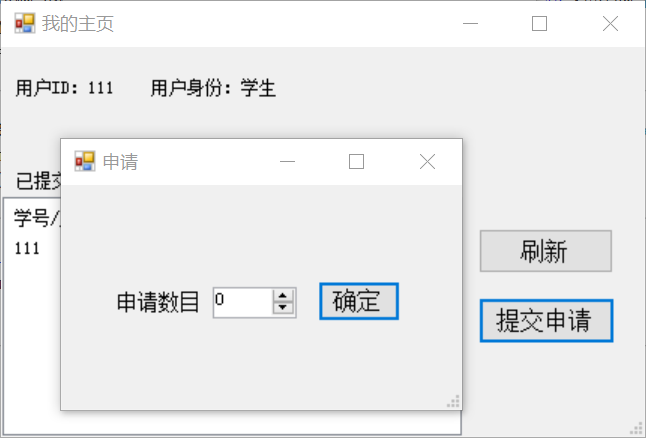


图 7-5 申请者主页

如图7-6所示，若登录用户为审批者身份，则页面还会给出待审批申请项数，列表框内显示待审批的申请数目和审批状态。用户先选择列表框中某一申请项，再点击“审核申请”按钮，会弹出审批页面，用户可以选择同意或者不同意该项申请。

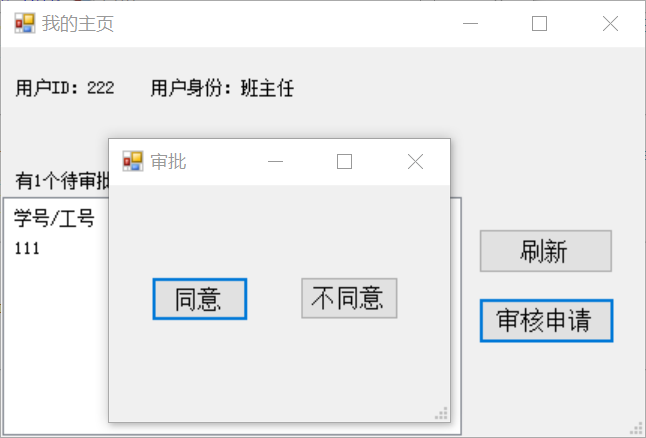


图 7-6 审批者主页