

系统设计文档

一、 引言

1.1. 编写目的

该文档主要是为了明确数据结构和系统结构,说明系统整体的概要设计,为下一步的详细设计做准备。也为之后系统的编码实现提供支持。

1.2. 背景

- a) 项目名称为基于 JAVA 的助学生软件。
- b) 本项目的提出者为龙行超,开发者为全体组员,该项目的用户是在校学生。
- c) 该项目运行于 PC 平台之上,需要无线局域网环境的支持。

1.3. 定义

- a) 导入课表:利用教务处自带的导出课表功能产生一个 pdf 文件,之后直接往软件里导入该 pdf 文件分析出正确的课表信息。
- b) 推送公告:从学校教务处里直接导入相关通知和新闻。
- c) 事务提醒:使用邮件的方式通知相关信息。
- d) 安排计划:将自己的计划写入系统,并设置提醒。
- e) 任务制定:与相似目标的伙伴,共同制定需要完成的任务内容。

1.4. 参考资料

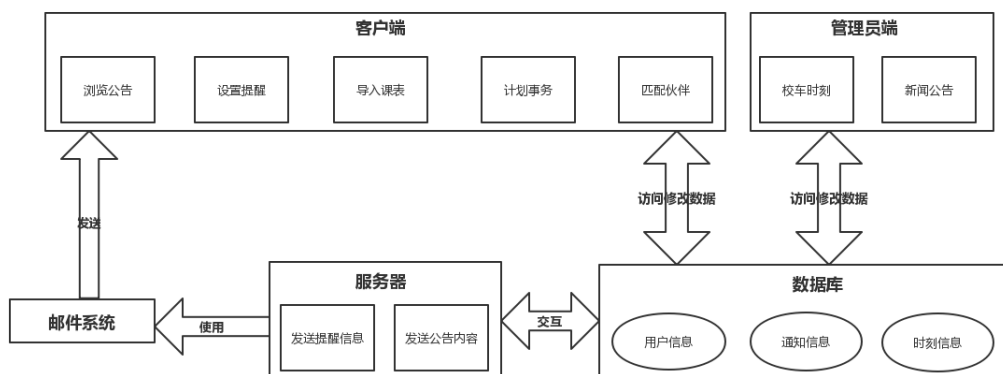
- [1]. Bruce Eckel . Java 编程思想(第 4 版). 机械工业出版社. 2007-6
- [2]. 几种网络通讯协议 (WIFI,WAPI,GSM,GPRS,CDMA) 概述 <http://www.doc88.com/p-687406832199.html>
- [3] Java 读取 PDF 文本内容 <https://blog.csdn.net/Eiceblue/article/details/82592566>
- [4]杨冠宝.阿里巴巴 Java 开发手册.电子工业出版社.2018-1

二、 总体设计

1.1. 系统设计原则

- a) 单一职责原则。各个子系统的职责不交叉,各个子系统的内部的类设计职责单一。
- b) 里氏替换原则。父类的方法子类一定要全部实现,也就是在调用父类的地方换用子类也一定能完成。
- c) 接口隔离原则:不同的业务逻辑拆分成不同的部分,避免不必要的实现。
- d) 迪米特法则:一个对象应该对其他对象具有最少的了解。
- e) 开闭原则:一个软件的实体如类、模块和函数应该对扩展开放,对修改关闭。

1.2. 系统框架设计



系统的总体框架设计如上，采用 C/S 架构，C/S 架构有如下优点：

1. 应用服务器负荷小。客户端与应用服务器之间只进行数据交换，而服务器上不对数据进行处理，极大减轻服务器的负担。
2. 充分利用客户端的处理能力。C/S 架构减轻了服务器端的压力，把绝大部分事务都放在了客户端，这使得客户端机器的能力被极大的利用，不会造成客户端资源流失。
3. 交互速度快。无中间组件，客户端和服务器之间直接相连，直接进行数据的交互，而不需要中间的组件对数据进行处理，从而其速率将会相当的高。

本系统分为五个部分，应用层由两部分组成，一是客户端、二是管理员端，而后台服务器由三部分组成，一是数据库服务器、二是逻辑服务器、三是邮件系统。应用层与服务器直接交互的是邮件系统和数据库服务器，应用层能够访问和修改数据库服务器上的信息，而邮件系统则通过邮件的形式与客户端单项交互。邮件系统与数据库通过中间的逻辑服务器相连，服务器访问与修改数据库中的信息，进而使用这些信息，借助邮件系统通知客户端。

以上是各组件之间的关系，接下来，细谈一下，各组件的作用：

1. 客户端：学生使用的软件，用于导入课表、设置提醒、查看新闻、计划安排、匹配伙伴等功能的逻辑实现，充分利用其客户端的优势，降低服务器的压力。
2. 管理员端：老师、及管理员使用的软件，用于录入校车时刻表和新闻公告，直接与数据库进行交互，使得速度得到提升。
3. 数据库：用于存储各式各样的信息，包括学生的基本信息、校车时刻表、公告和新闻等信息，是整个系统的信息中心，是限制性能的最大瓶颈。
4. 服务器：此服务器用来遍历数据库中的每条提醒信息，并借助之后的邮件系统，将信息推送给客户端——学生方，这里的设计采用并行处理的方式以便提升速度。
5. 邮件系统：负责邮件的发送，借助第三方的邮件服务器来组建该系统，如：腾讯 QQ 邮件服务器、网易 163 邮件服务器、网易 126 邮件服务器。

采用上述框架有如下的优点：

1. 耦合度低、内聚性高。各个系统之间交互少，没有不必要的系统间的交互，使得系统的各个部分耦合度低；各个部分各司其职，分工明确，职责专一，具有很好的内聚性。
2. 易于团队开发。各个系统之间耦合度低、内聚性高，易于团队的分工，也便于项目的整合，从而提高团队开发效率。
3. 便于维护。各个系统之间除了数据，并无其他交互的地方，这就意味着，对于

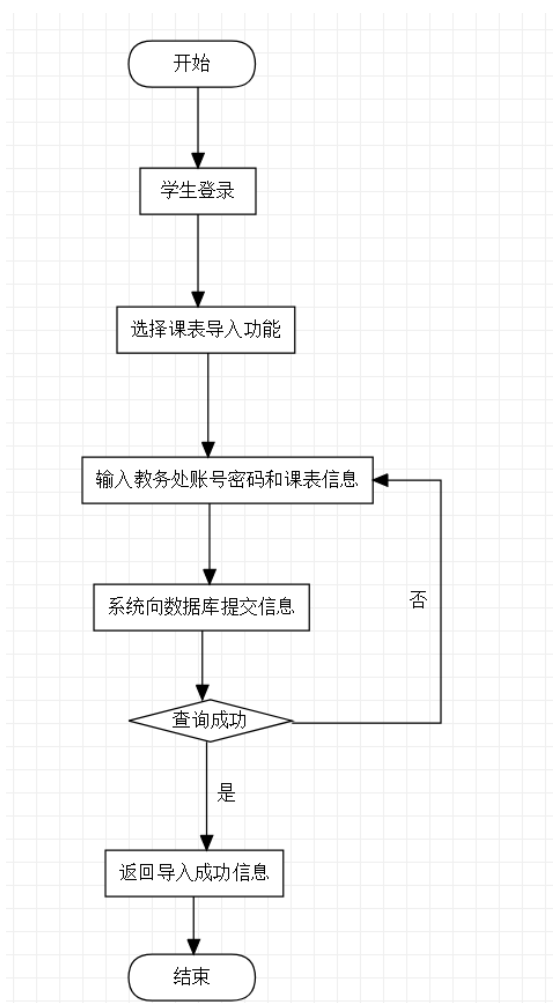
一个子系统的修改，并不需要对其他系统进行修改，使得维护的难度大大降低。

4. 软件复杂度低。系统没有不必要的耦合，内部的功能明确，这便使得该软件的复杂性降到了最低，便于开发此软件。

总之，软件总体设计框架如上述那样，给我们的团队开发此项目带来了极大的好处，不仅其设计简单、易于开发，还要求强的服务器，这是最符合我们团队现状的软件开发框架。

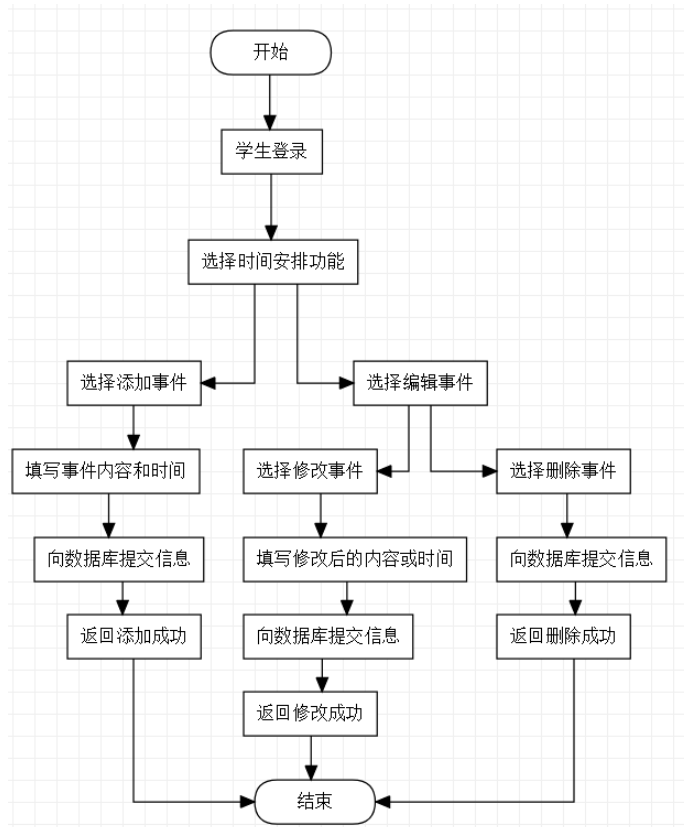
三、 系统功能设计

3.1 学生导入课表



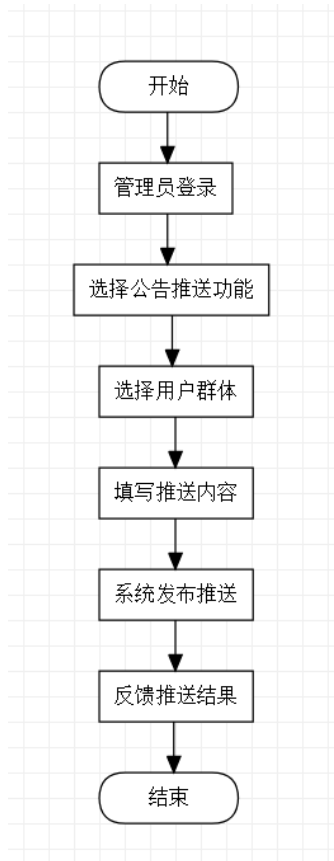
用户可以在课表导入功能中简单的输入自己的教务处账号密码后，完成对课表的导入。系统则利用教务处自带的导出课表功能产生一个 pdf 文件，之后直接往软件里导入该 pdf 文件分析出正确的课表信息。而不是要求用户去一张张的复制粘贴过来或者自行填写。

3.2 学生安排时间



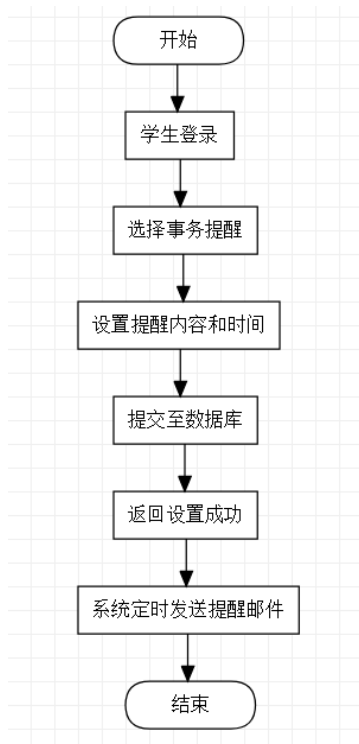
安排时间功能采用先选择时间，后填写事件的模式，即用户点击可视化的时间轴来插入，修改或删除相应事件，对应着系统对数据库的插入，更新，删除操作。

3.3 推送学校公告



推送学校公告功能需要管理员手动从学校教务处里直接导入相关通知和新闻, 然后选择需要进行推送的用户群体, 系统会在用户使用端首页进行推送。

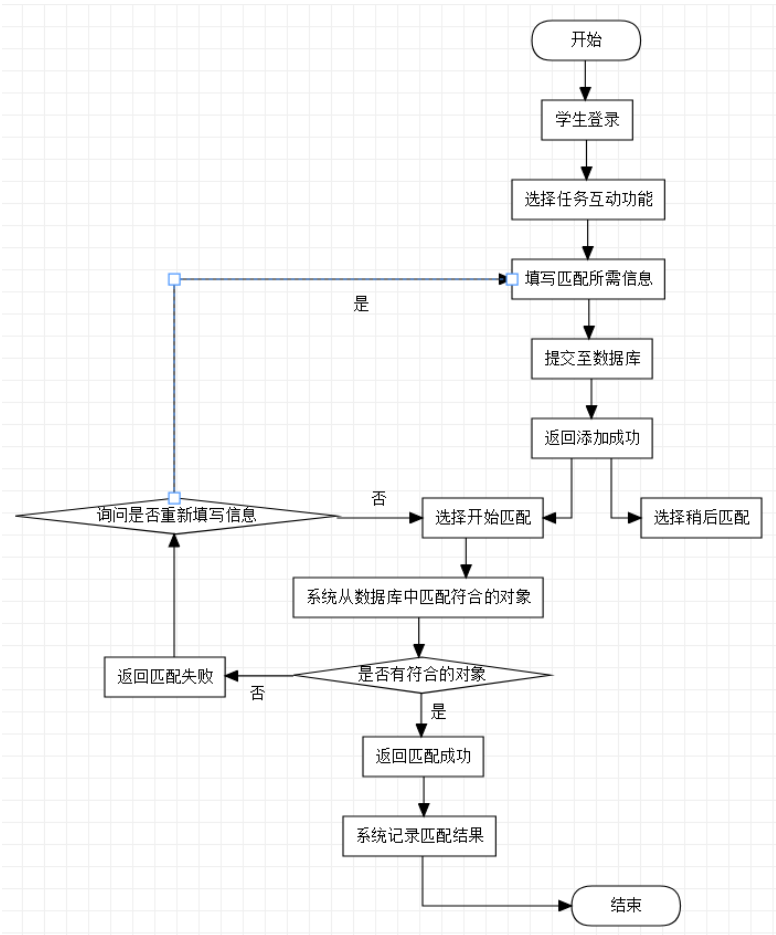
3.4 学生事务提醒



事务提醒功能采用传统的软件内部消息通知和邮件的方式来提醒, 提前三十分钟提醒,

之后每隔十分钟提醒一次。用户只需自行设置需要提醒的时间和内容，系统会根据时间向用户发送预设内容的邮件和消息。

3.5 学生任务互动

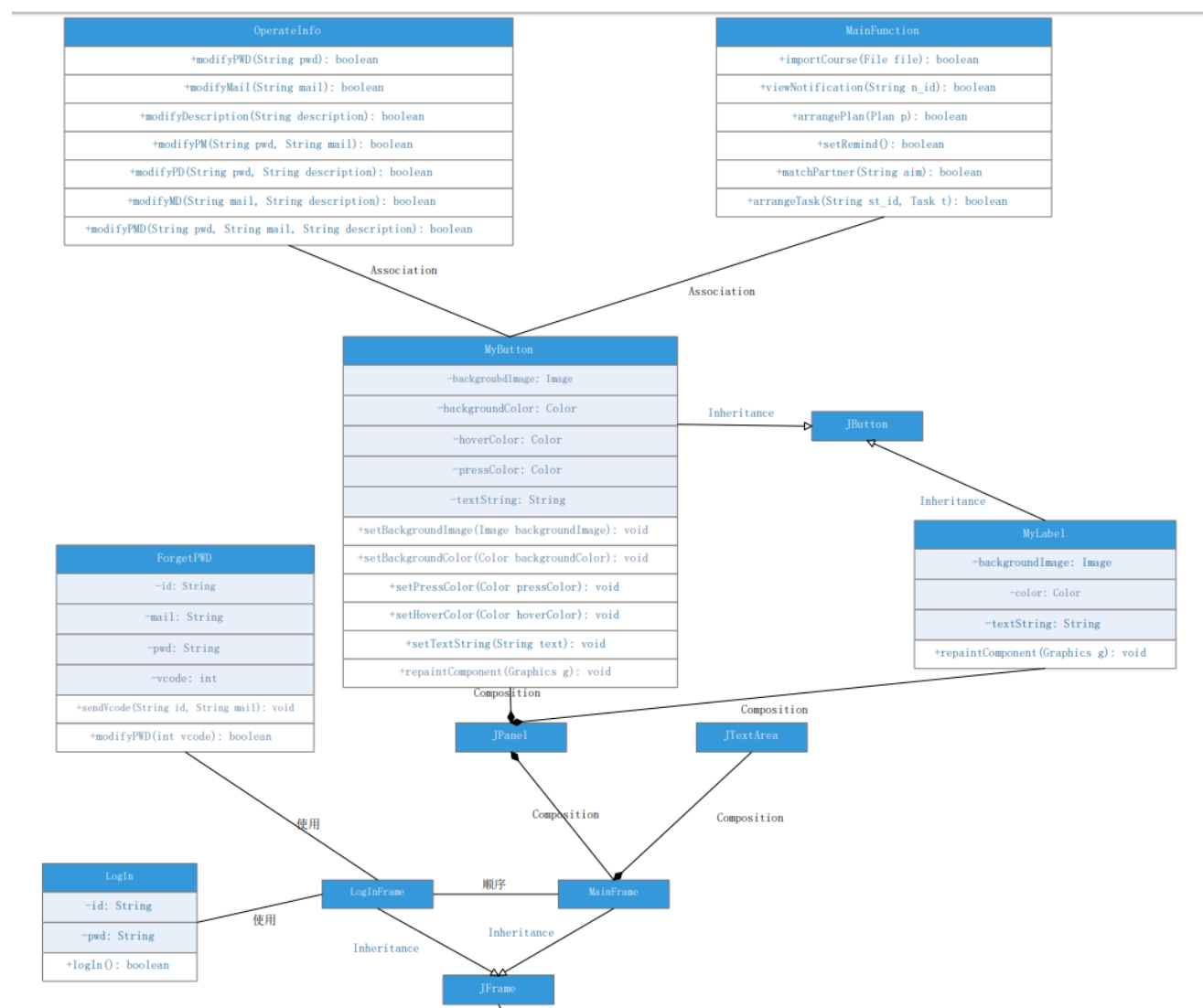


任务互动功能会引导用户根据自身的实际情况，填写自己的目标等信息，系统会根据关键字或者时间等在数据库中进行配对，并将结果反馈给用户。

四、 类设计

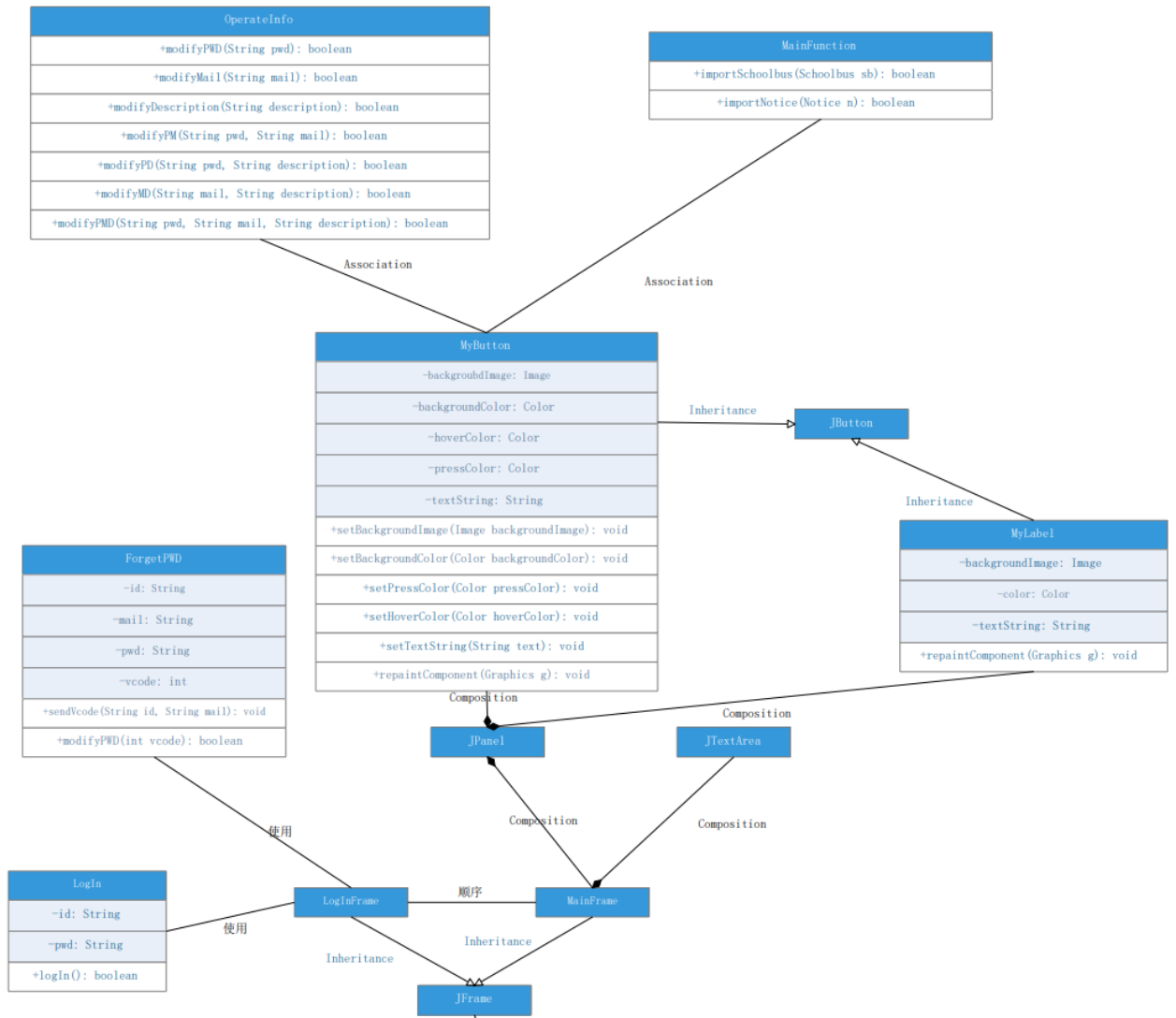
类分为三部分进行设计，分别是客户端、管理员端、服务器。各部分的类图如下（图可能看不清楚，因为类中的数据太多，可访问[如下网站](#)：[类图](#)）：

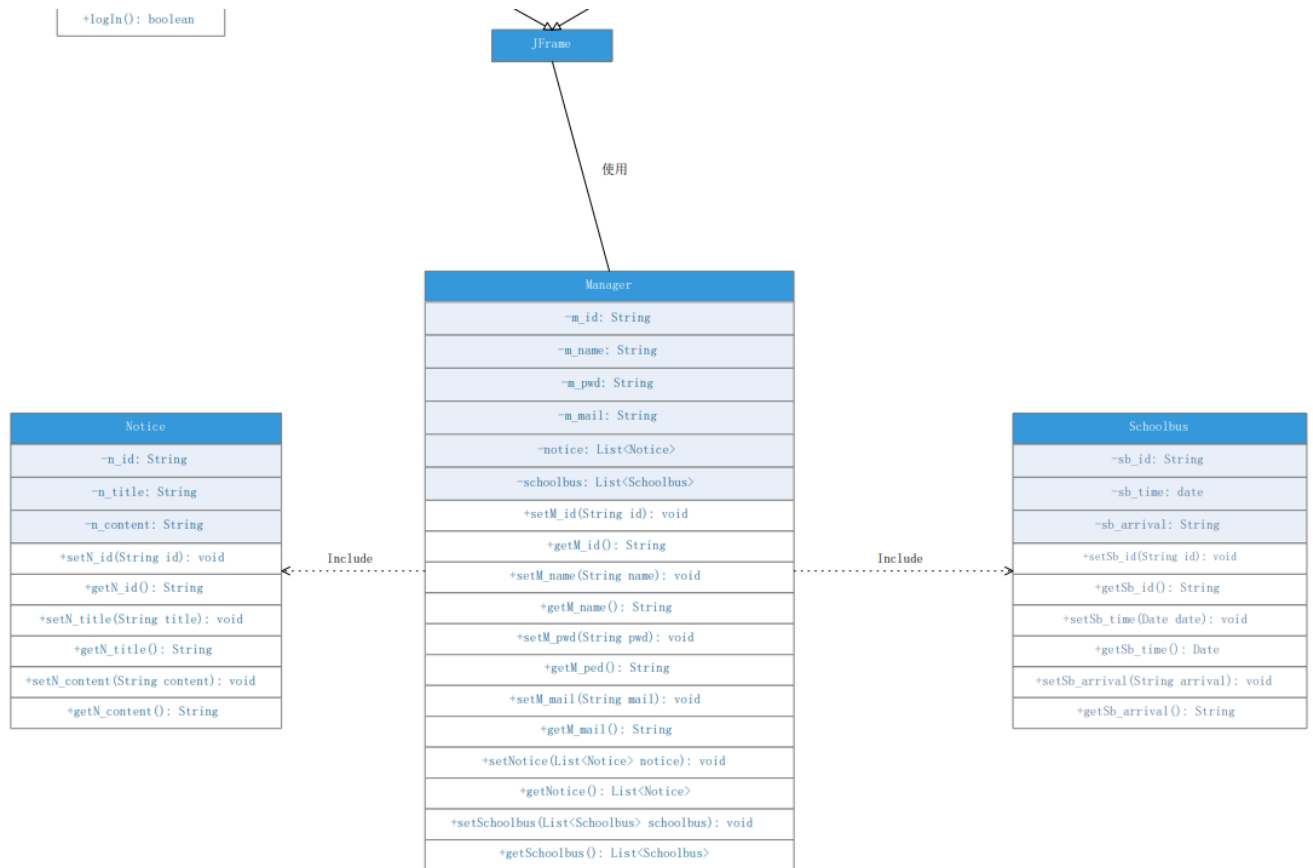
4.1 客户端



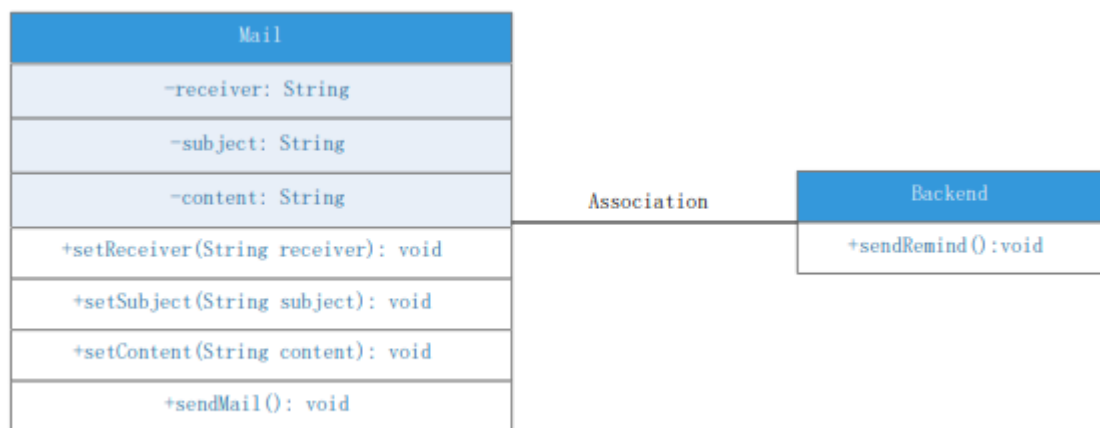


4.2 管理员端



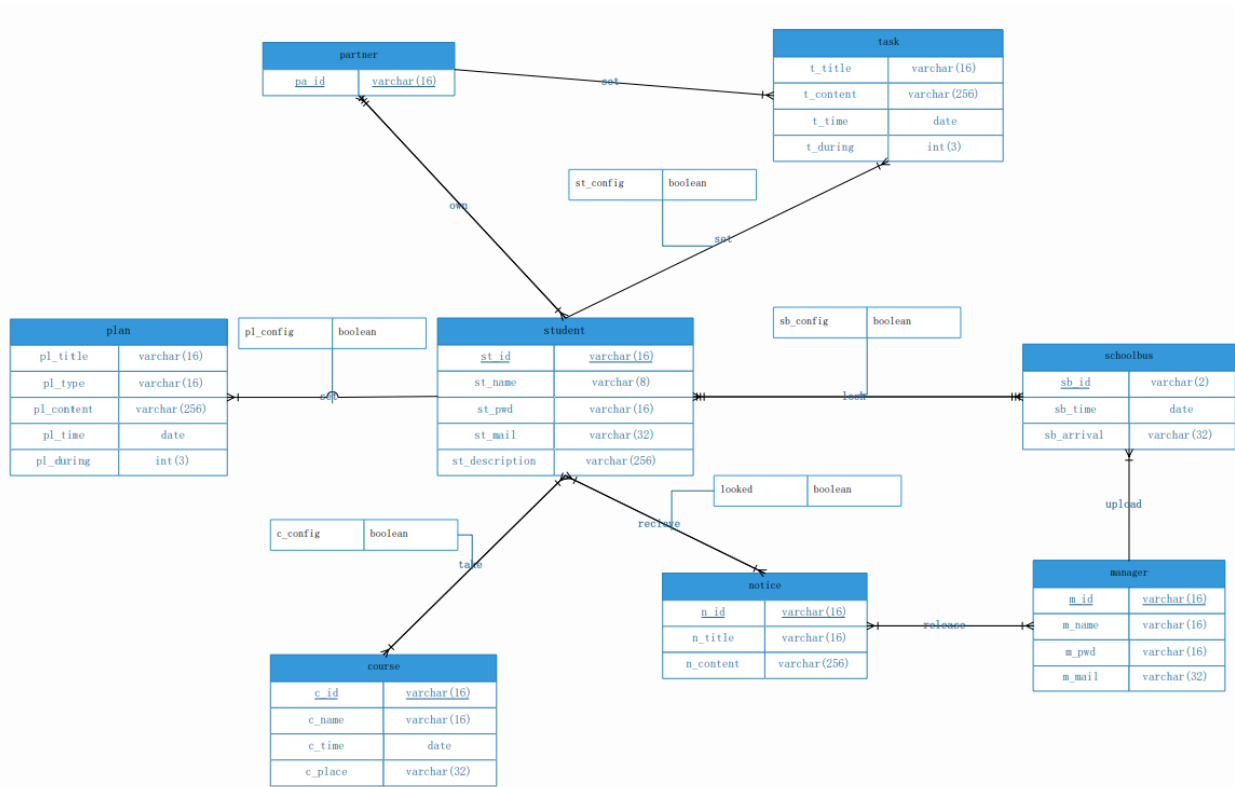


4.3 服务器端



五、 数据库设计

er 图设计如下（如看不清，请点击如下链接：[ER 图](#)）：



转换为关系表如下：

student				
字段名	数据类型	长度	说明	描述
st_id	varchar	16	不空，主键	学生编号
st_name	varchar	8	不空	学生姓名
st_pwd	varchar	16	不空	学生密码
st_description	varchar	32	不空	学生简述
st_mail	varchar	256	不空	学生邮箱

plan				
字段名	数据类型	长度	说明	描述
st_id	varchar	16	不空，外键	学生编号
pl_title	varchar	16	不空	计划标题
pl_type	varchar	16	不空	计划类型
pl_content	varchar	256	不空	计划内容
pl_time	date		不空	计划开始时间
pl_during	Int	3	不空	计划持续时间
pl_config	boolean		不空	有无计划

course				
字段名	数据类型	长度	说明	描述
c_id	varchar	16	不空，主键	课程编号
c_name	varchar	16	不空	课程名字
c_time	date		不空	课程时间

c_place	varchar	32	不空	课程地点
---------	---------	----	----	------

c_take				
字段名	数据类型	长度	说明	描述
c_config	boolean		不空	有无设置提醒

notice				
字段名	数据类型	长度	说明	描述
n_id	varchar	16	不空, 主键	公告编号
n_title	varchar	16	不空	公告标题
n_content	varchar	256	不空	公告内容

receive				
字段名	数据类型	长度	说明	描述
looked	boolean		不空	有无查看

manager				
字段名	数据类型	长度	说明	描述
m_id	varchar	16	不空, 主键	管理员编号
m_name	varchar	16	不空	管理员名字
m_pwd	varchar	16	不空	管理员密码
m_mail	varchar	32	不空	管理员邮箱

schoolbus				
字段名	数据类型	长度	说明	描述
sb_id	varchar	2	不空, 主键	校车编号
sb_arrival	varchar	32	不空	到达地点
sb_time	date		不空	出发时间

sb_look				
字段名	数据类型	长度	说明	描述
sb_config	boolean		不空	有无查看

partner				
字段名	数据类型	长度	说明	描述
pa_id	varchar	16	不空, 主键	好友编号

task				
字段名	数据类型	长度	说明	描述
st_id	varchar	16	不空, 外键	学生编号
pa_id	varchar	16	不空, 外键	好友编号
t_title	varchar	16	不空	任务标题
t_content	varchar	256	不空	任务内容

t_time	date		不空	任务开始时间
t_during	Int	3	不空	任务持续时间
pa_config	boolean		不空	好友是否设置提醒
st_config	boolean		不空	学生是否设置提醒

六、 界面设计

6.1 界面设计原则

(1) 系统的界面风格统一，使用墨刀对单元格、按钮、下拉列表、文本框都进行统一的规格化，页面布局采用上方导航栏下方功能页的页面布局。在内容填充中，验证每个条目的数据合法化。如果出现异常或错误，将使用统一的错误页面和易于理解的提示语言来描述异常或错误。

(2) 对于用户操作来说，越容易、越简便越好，在系统设计和开发过程中我们将体现的友好操作页面，软件具有较高的可用性。

(3) 本系统采用同步和异步的方式进行数据交互，异步操作可以方便用户在页面操作过程中与数据库中的数据交互。同步操作可以使用户在提交页面时实时查看和修改提交的内容。十分合理、高效且有序。

(4) 系统提供在操作过程中根据输入项和功能来提示的功能来帮助用户更好的使用和操作系统。

(4) 系统界面具有十分清晰的布局和文字描述，方便了程序的开发和维护。

6.2 界面层设计技术

(1) 在 java swing 页面中通过页面文本框组件、下拉列表组件、单选按钮组件、多选按钮组件、按钮组件等组件对页面进行设计和实现。

(2) 页面采用后台同步技术实时访问系统，也支持异步技术访问页面中的系统，从而在页面与系统交互后获取相关内容。

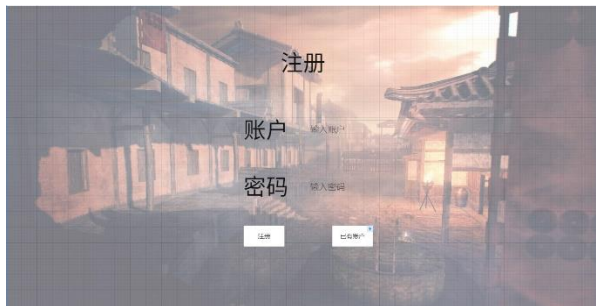
(3) 使用墨刀等软件对页面进行加工和整合，使用户更方便的操作和使页面程序可读性更高。

6.3 界面功能实现

用户登录页：



用户注册页：



用户通知页：



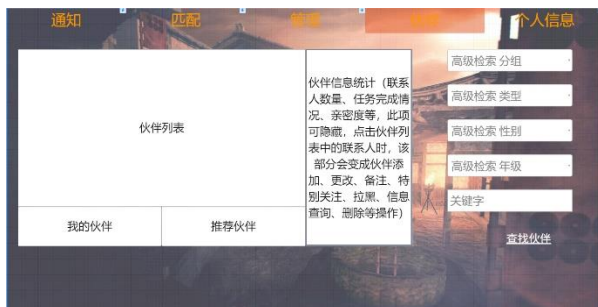
用户匹配页：



用户管理页：



用户伙伴页：



用户个人信息页：



用户课表管理页：



用户事务管理页：



用户计划管理页：



用户提醒设置页：



用户今日日程页：



管理页登录页：



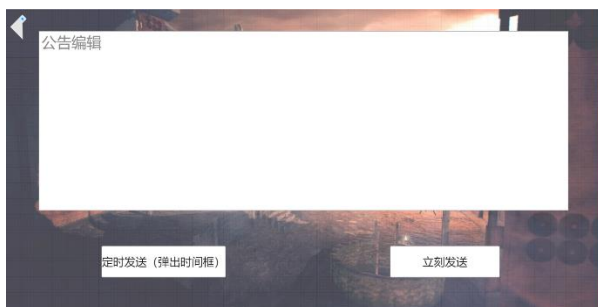
管理员主页：



管理员校车时刻表管理页：



管理员公告管理页：



管理页个人信息页：



七、 总结

我们时常考虑，什么是真实，什么是虚幻。真实就是可观可感可知之物吗？我们认为不是。而虚幻就是离散乖离甚至不存在之物吗？我们也无法认同。在真实的日常学习生活中，我们却时常感觉到一种缺失，一种不和谐。我们不知道，作为学生，需要的但是却不存在的是什么，我们想探寻真物，但是又执着于眼前的惆怅，我们四处搜集资料、访问群众，但仍十分迷惑。这次，我们想：也许从我们最熟悉的软件中，能找到答案。

软件是人类生活不可或缺的一部分，自上个世纪，个人计算机出现以来，软件就成为了人类最好的伙伴。不，甚至更早，早在硬件出现的那一刻，软件就随之现世了。我们想着软件开发，想着我们迫切需要的，我们如此想着，所以我们如此做了。一款新颖且高效的助学生软件，不追求潮流，只追求卓越，低调但不低端，这正是我们所希望看见的东西。

我们开发的这款软件，继承了前人软件开发中积淀下来的经验，毕竟软件开发的各种原则，23种设计模式，我们早就烂熟于心。这不是尝试，这是我们应做到的，对自己心中疑问的回答。艺术的使命是崇高，时间艺术的目的是消除日常生活的贫乏。我们希望这款软件，足够强大，足够细心，也足够温暖，我们看到学生们最需要的，软件做到学生最需要的，就够了。于是，良好的架构和优秀的设计，是重中之重。

我们采用了经典但不失内涵的 C/S 架构，在个人设备性能不断提高的今天，这是一种历久弥新、不断展现出强大生命力的框架。我们创造性地设计出了，兼顾高可用性、高易用性和高可修改性的用户界面，在频繁的人机交互中，冷冰冰的机器，也能将我们内心的温暖，传达给每一个学生用户。

我们不缺乏真实，但我们缺少真理。真理是需要探求的，我们做了一款简单但不低端的软件，希望能贴近生活，贴近真理。我们有充分的理由去相信，它能给用户带来最美妙的体验