**基于Spring Boot的宿舍管理系统 系统分析与设计**

**「~~（１）题目可自拟~~，或由**

**（２）教师统一公布一批题目」**

**班级 周三**

**序号 63**

**学号 20251104157**

**姓名 郑圳辉**

***封面设计，由教师来决定。***

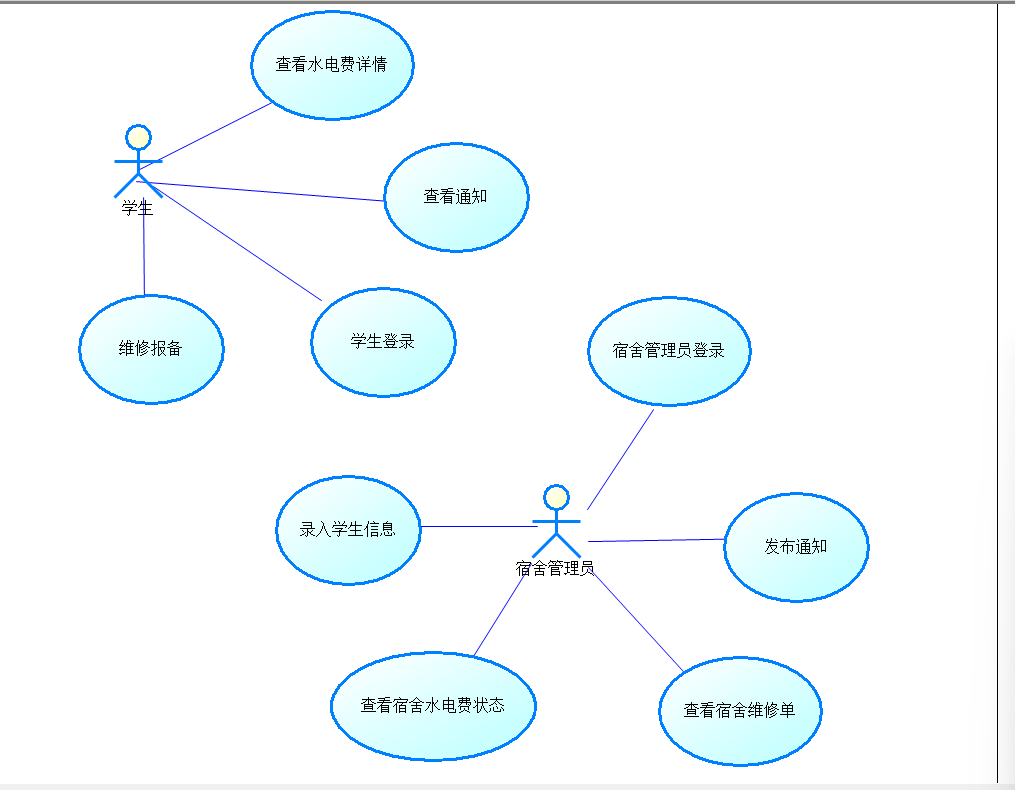
**第1部分-用例模型**

* 简要文本描述系统需求。「参考第10章 S10.1节」

首先这个系统是宿舍管理系统， 既然是管理宿舍， 肯定有管理者，和被管理者， 所以这个系统有两个角色， 一个角色是学生， 另一个角色就是宿舍管理员， 在用例方面

一个宿舍每个月都有水电费要交， 这个系统可以提供一个接口给学生查看这个月的水电费详情， 而对于管理者主要是查看水电费的缴费情况， 同时，当宿舍中有基本生活设备损坏的时候， 系统能提供给一个接口给学生进行报备， 而管理员也需要一个接口可以查看报备情况， 对报备的宿舍安排维修人员。 更重要的是这个系统是一个信息发布平台， 宿舍管理员可以在这个平台发布通知， 而学生可以在这个平台上查看通知。 此外除了这些系统基本的功能需求之外， 系统还需要给学生提供登录， 管理员提供录入学生信息， 及管理员登录的功能。

* ***用例图***-要求包含全部用例，在图中要表明用例关系，参与者要说明身份。「参考 图10.3」



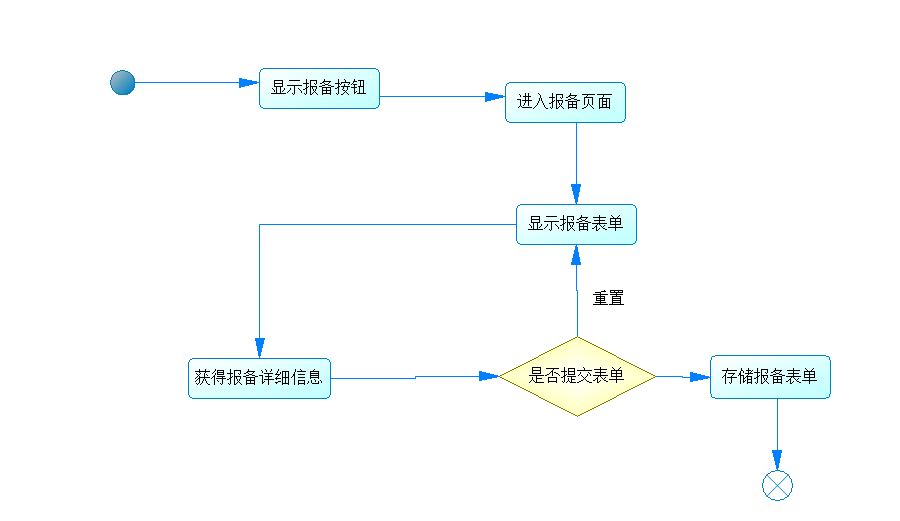
* 从用例图里**挑出一个典型用例（比如 Use Case A）**（复杂度中等）－

我们用用例图中的维修报备用例

* 撰写**其-UC A**用例文档 「参考表10.3」

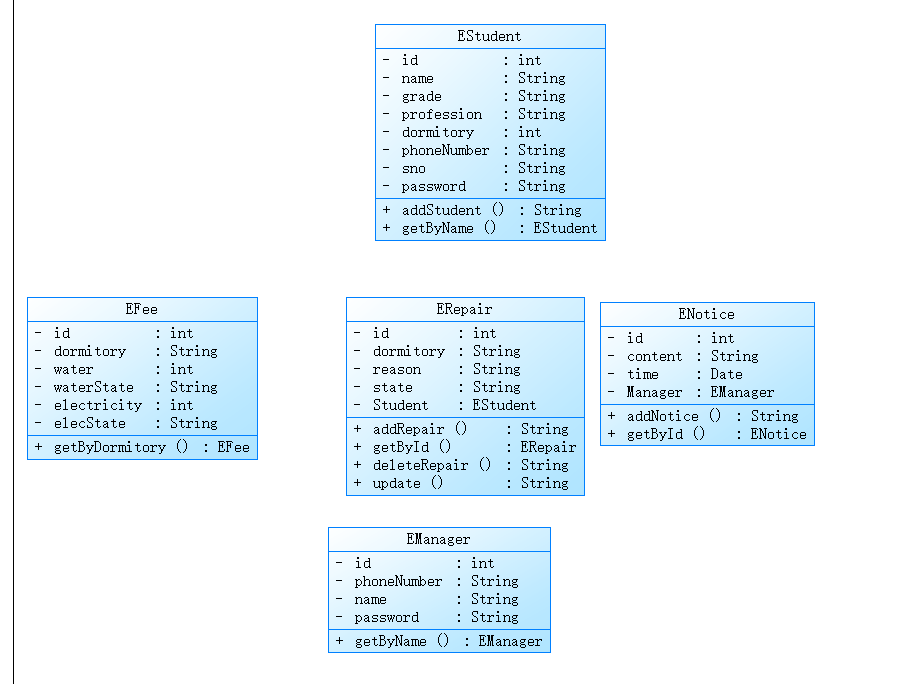
|  |  |  |
| --- | --- | --- |
| 编号 | 用例说明 | 动作 |
| 1 | 当维修报备按钮显示在 屏幕上， 学生通过点击决定维修报备，维修报备用例开始 | 显示报备按钮，  进行报备页面 |
| 2 | 系统要求学生通过文本输入报备的具体信息，报备宿舍， 报备原因， 选择报备状态。 | 显示报备表单 |
| 3 | 学生选择提交按钮，将详细的报备表单提交给宿舍管理员 | 获得报备详细信息 |
| 4 | 系统将报备表单存储到数据库 | 存储报备表单 |
| 5 | 学生未提供报备所要求的信息， 这时系统显示出错页面，要求学生输入所缺信息 | 获取报备详细信息  显示报备表单 |
| 6 | 学生选择reset功能回到空白的报备表单，系统允许学生重新输入 | 显示报备表单 |

* 绘制**其-UC A*活动图*** 「参考图10.5」

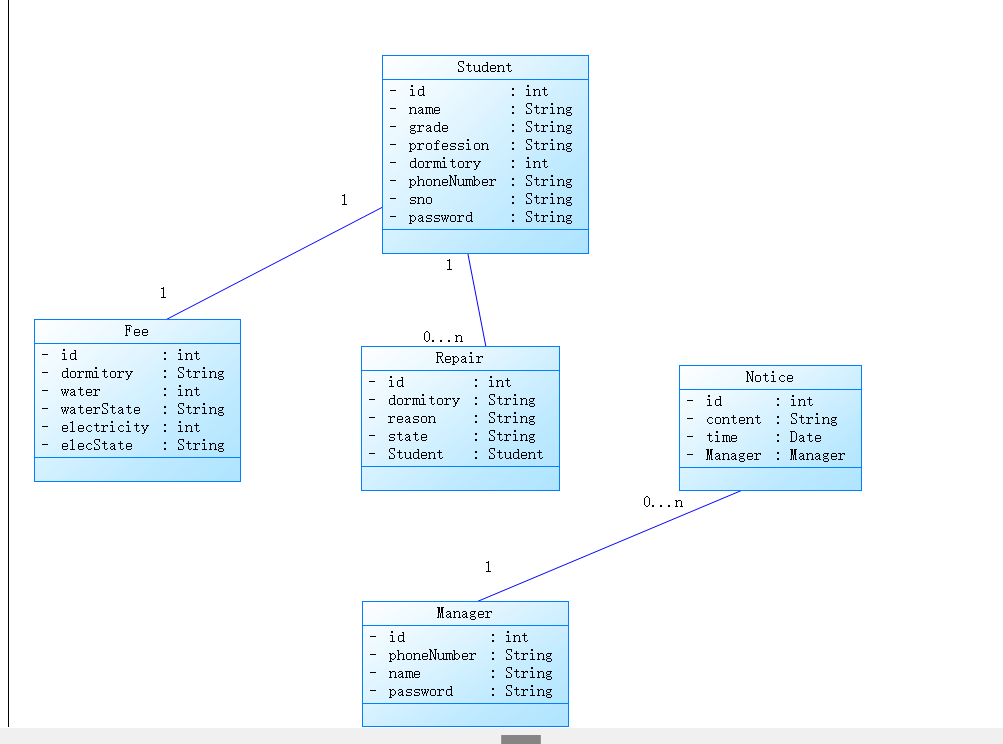


**第2部分-类模型**

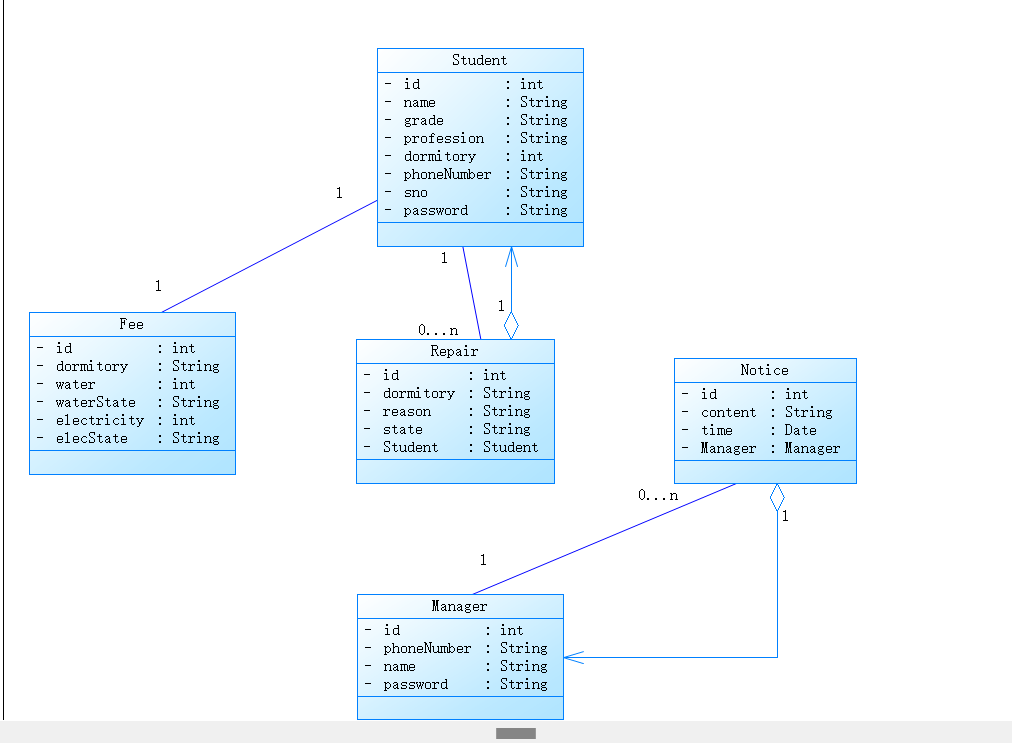
1. 列出本系统的实体类(属于 Entity层)（需持久化的类）「参考 图10.6」（注：图10.6 是类的集合，并非 ***类图模型***）



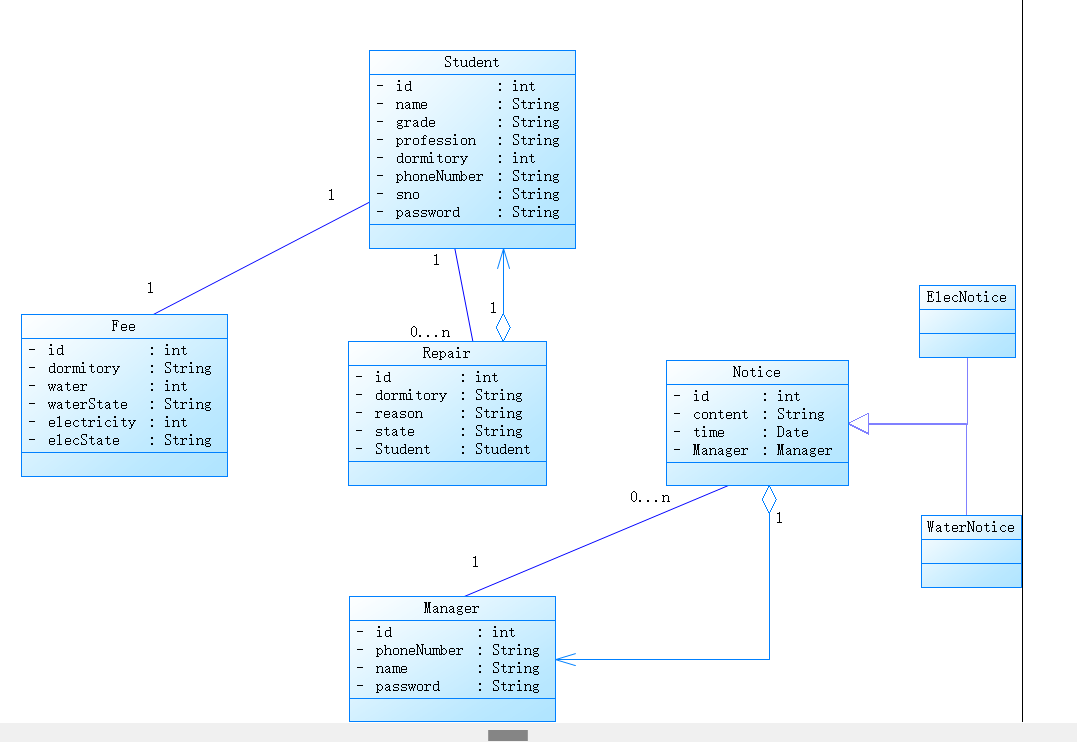
1. 为实体类中的***关联关系***建模「参考图10.8」



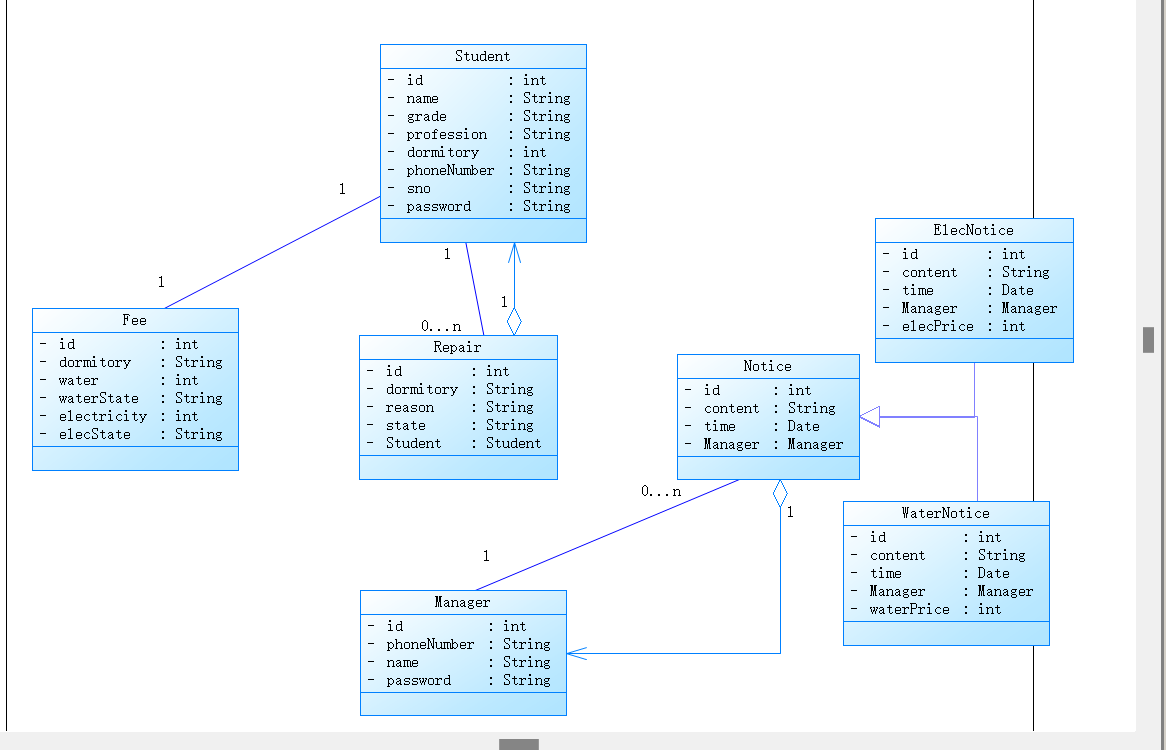
1. 为***聚合关系***建模「参考图10.9」



1. 为***泛化关系***建模「参考图10.10」



1. 建立***全部实体类的（分析 / 基础）类模型***「参考图10.11」，并对类模型进行简要解释。（本设计**禁止**直接从项目（1）跳到项目（5）！）



从关联上来说， 一个学生可以维修报备很多次， 一个学生只有一个宿舍， 当然也只有一张水电费表， 然后一个宿舍管理员可以发布多次通知。

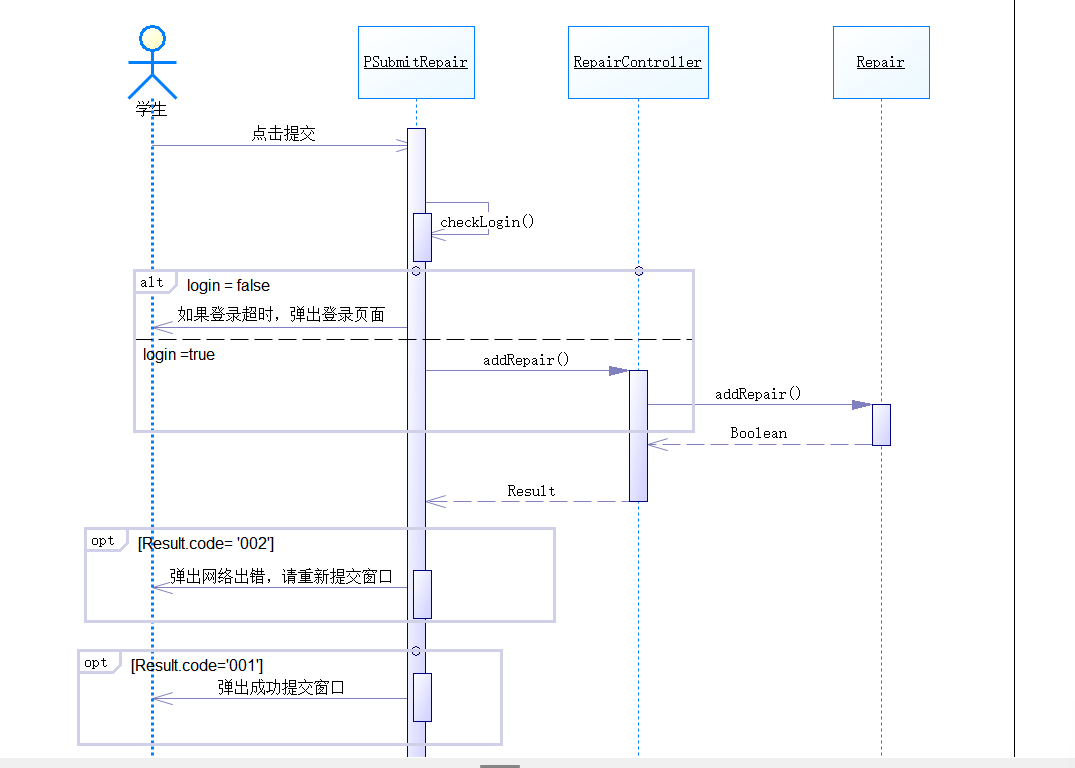
从聚合关系上来说， 当一个通知发布之后， 他需要一个发布者这个属性， 所以它聚合了一个宿舍管理员， 而维修单同样需要一个宿舍的学生当负责人， 所以一个维修单就聚合了一个学生类。

从泛化关系上来说， 将通知类改成抽象类， 它具有两个具体的子类， WaterNotice和ElecNotice。 Notice和宿舍管理员直接关联， 通知可以是水相关的通知， 也可以是电相关的通知， 比如说停电通知， 或者是停水通知。

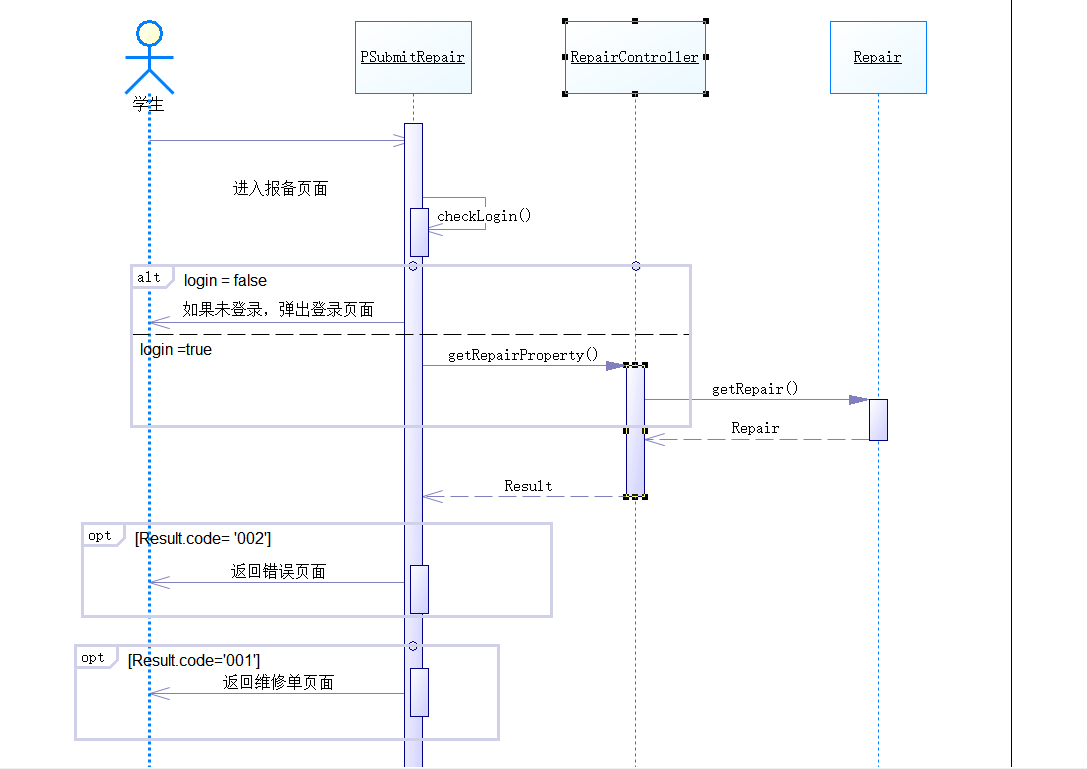
**第3部分-交互模型**

* 从第1部分的-***活动图模型***里找出***2个动作（Action,可视为用例的一个局部）***，分别建立它们的交互顺序图模型（或通信图模型） 「参考图10.14」

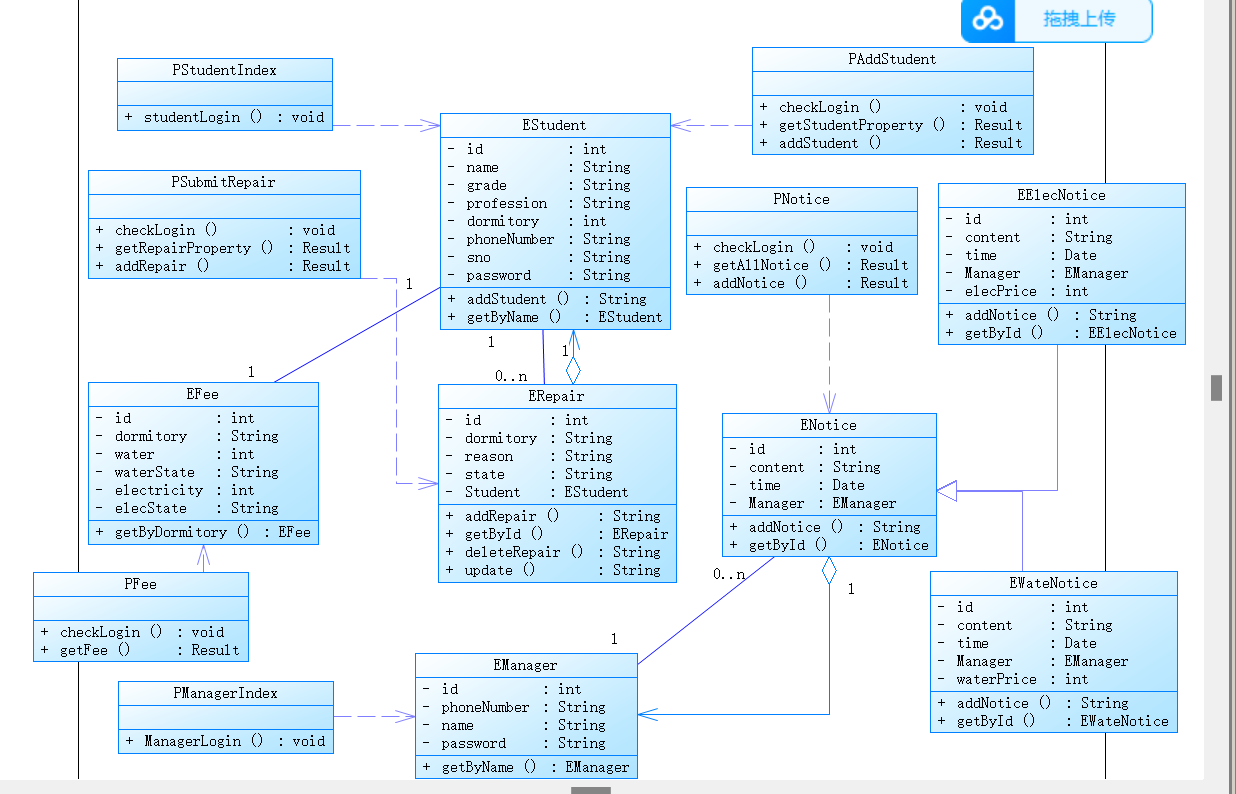
存储报备表单



显示报备表单



* 依据***交互图模型***为相关的类（指出现在交互模型里的对象所属的类）增加操作，操作包括：操作名，参数列表，返回类型。「参考图10.16，或图3-15」,得到本系统的***完整类模型***。并对类模型进行简要解释。



当除了首页的Page类对象， 得到请求消息， 都会检查是否登录， 或者登录超时即checkLogin() ,若是没有登录或者登录超时， 会返回学生或者宿舍管理员的首页进行登录，

当登录完成后才可以完成个Page页面的操作，

然后关于新增的操作，

（1）学生实体类增加了， 录入学生的功能和登录的操作。

（2）水电费账单则是查询具体水电费详情的操作。

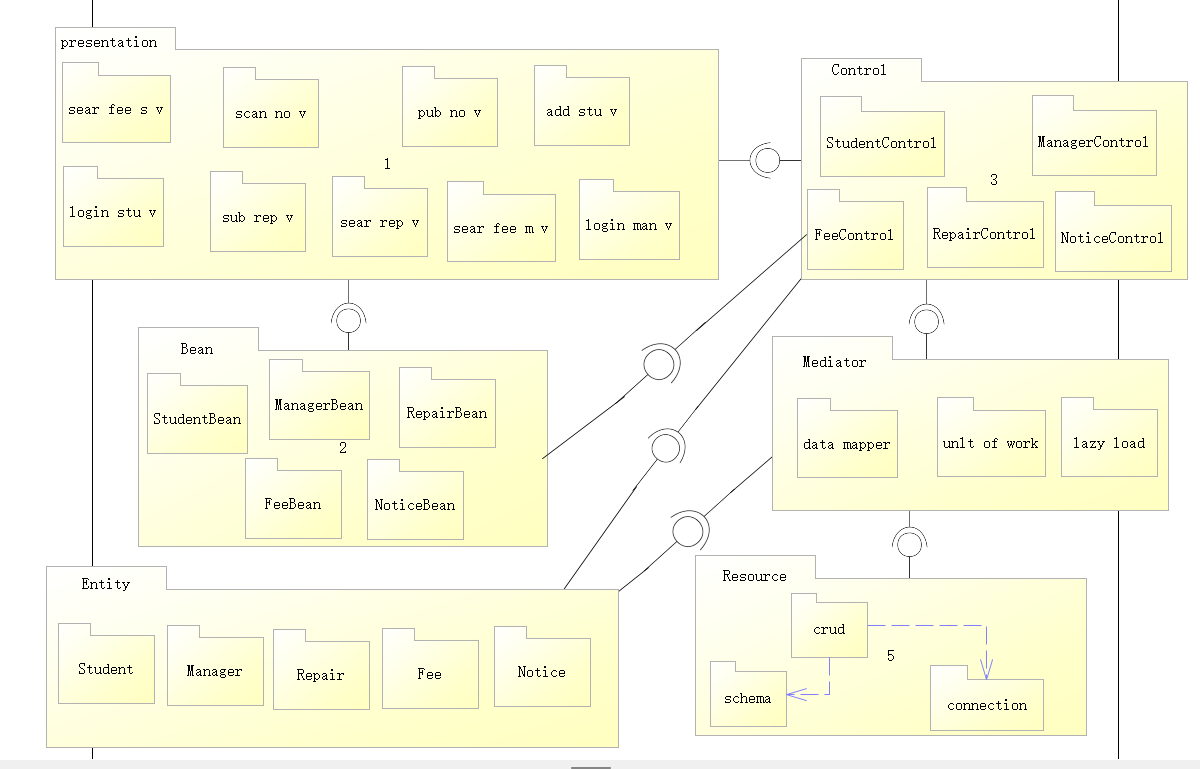
（3）宿舍管理员则是增加了登录的操作。

（4）维修单的实体类增加了添加维修单， 更新维修单， 查询维修单 ，删除维修单的操作

（5）通告的实体类增加了发布和查询的操作。

**第4部分-实现模型（这里的“实现”不同于对一个 Algorithm的实现）**

* 设计本系统的基于PCBMER体系结构的***包图模型***。 「参考图10.20」，并要求***对各个包的职责加以说明***；(加上你自己的理解)



完成基于springboot的宿舍管理系统最好的方法是“模仿系统”， 本系统需要给学生提供提交维修单， 查询水电费详情， 查询通知。 需要提供给管理员查询维修单， 查询水电费状态 ， 发布通知，录入学生信息， 所以整个系统有七个功能，还有学生和管理员的登录功能（首页）， 也就需要9个独立的GUI窗口，因此我们就可以创建九个表示层的包，考虑到我们表示层的包多， 实体类却少， 视图中所需要的数据从StudentBean包， FeeBean包，ManagerBean包， NoticeBean包， RepairBean这五个bean包提供（由control子系统创建）， 考虑到业务应用和实际需求，如果每个业务都要划分一个Control包， Control包未免太多而且臃肿， 所以 这些实体类和控制类都被按照实体类分为了五个包，

即五个Control，

1. NoticeControl， 它提供了notice类相关的所有服务。 包括管理员的发布通知的接口， 和学生的查询通知的接口，，
2. ManagerControl,他提供了管理员登录的服务。
3. RepairControl,它提供了维修单的相关的所有服务， 包括管理员查询所有维修单的状态， 学生提交维修单。
4. StudentControl, 它提供了学生相关的所有服务， 包括了管理员录入学生信息， 和学生登录功能。
5. FeeControl ，它提供了水电费账单相关的功能， 包括了管理员和学生的查询。

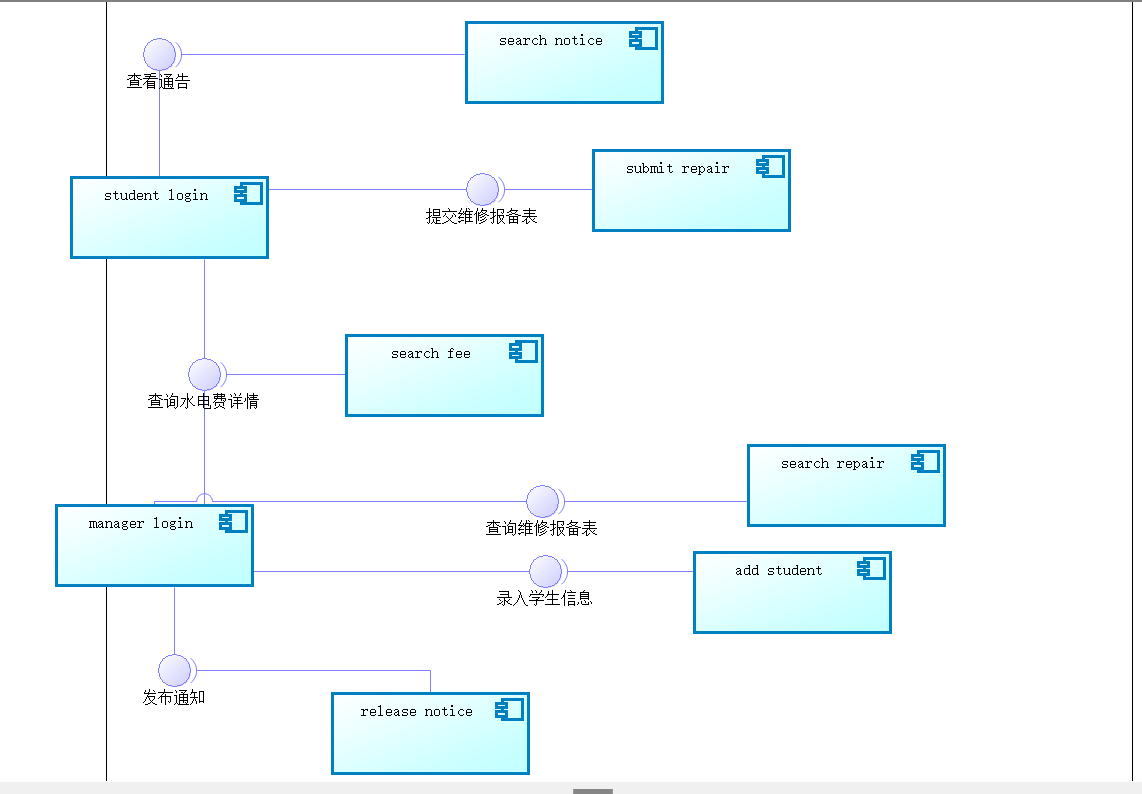
除此之外 ， 我们需要一个包来分配OID 到实体对象并维护OID映射， 我们将这个包命名为identity map包。

而对于中介者子系统， 我们由三个基本的中介者包， data mapper ,lazy load 和 unit of work .

最后需要一个或多个资源包， 主资源包名为crud， controller通过crud包与数据库表进行通信， crud包依赖另两个包， connection 包和schema包。 connection包中的类负责建立连接， schema则包含数据库模式对象的当前信息。

* 本系统的***构件图模型*** 「参考图10.21」，并***对各个构件的职责进行注解说明***。

\*：**包** 不同于 **构件（子系统）**，(见 S.3.6.1节)



如上图所示， 本系统有8个构件，

当学生或者管理员进入本系统， 首先看到的就是各自的登录页面， 所以有各自对应的登录子系统， student login , 和 manager login , 然后对于管理员来说 ，登录后就有四个选项， 分别对应四个不同的子系统， 第一个就是发布通知的子系统，

Release notice , 第二个就是录入学生信息的子系统， add student , 第三个就是查询宿舍水电费提交详情的子系统， search fee ,第四个就是查询维修报备表的子系统， search repair,

另一方方面对于学生来说， 成功登录后， 就有三个选项， 第一个就是查询上个月的水电费， search fee子系统这个子系统和管理员和公用的， 第二个就是提交维修报备表， 即submit repair 子系统， 最后就是查询通告， search notice 子系统，