



校园物业报修系统

分析与设计

成员：

卢建业 13331183

龙德成 13331180

刘健诚 13331169

罗双奇 13331192

周基源 13331370



注：本软件同时也是系统分析与设计的课程作业，小组成员与系统分析与设计的成员重叠，组长相同。

内容目录

- 第 1 章 架构设计.....2
 - 1.1 架构描述.....2
- 第 2 章 子系统及接口设计.....5
 - 2.1 物业报修系统架构图.....5
 - 2.2 子系统及接口设计描述.....6
 - 2.2.1 学生注册子系统.....6
 - 2.2.2 学生管理子系统.....7
 - 2.2.3 报修单管理子系统.....8

第 1 章 架构设计

1.1 架构描述

本系统采用 MVC 三层架构的系统设计框架，这三层分别为：表示层，控制层以及实体层。该三个分层结构具有明晰的依赖关系，表示层依赖于控制层，控制层调用实体层。

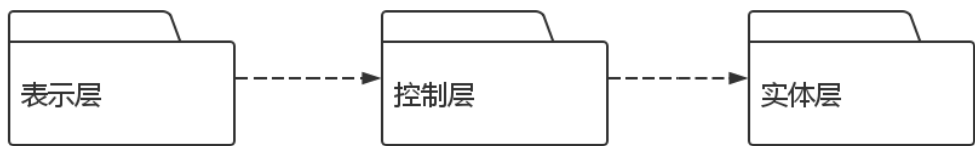


图 5 – 架构图

本项目进一步对上结构做细化，情况如下：

（1）表示层，该层是用户（包括学生与系统管理员）与系统进行交互的地方，负责显示相应的界面，获取用户的输入，做出相应的响应。**在源码中，对应的是 webapp 文件夹下的代码，它包括了控制显示效果的 html，css，jsp 代码以及图标文件。**

（2）控制层，负责执行整个系统的业务操作，控制整个系统的运行。首先系统的控制层会接收用户请求消息，然后调用实体层数据，接着负责执行业务操作，最后将操作的结果传输到表示层显示给用户。控制层实现技术为 SpringMVC 的控制器，控制器的工作流程图如下：

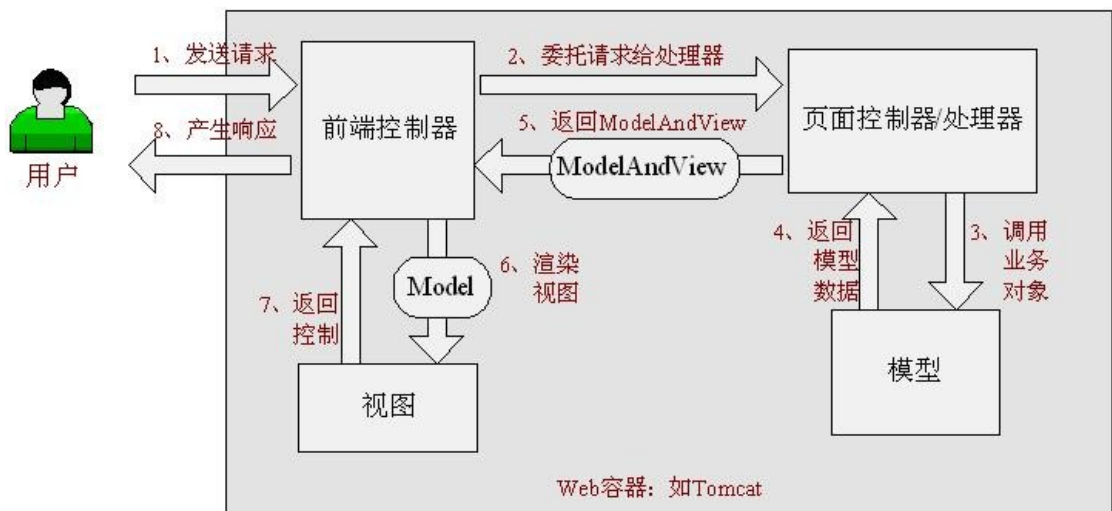


图 6 – 控制层的流程图

在源码中，对应的是 interceptor，service 文件夹下的代码，完成用户发送的各种数据请求，以及 dao，dto 文件夹下的代码，用来并调用下层的实力层接口。

（3）实体层，实体层封装了视图所需的所有数据，以 JAVA BEAN 的形式表示。**在源码中，对应的是 entity 文件夹下的代码，定义了各个实体类的属性和操作。**

第 2 章 子系统及接口设计

2.1 物业报修系统架构图

我们按照系统中所有的类的位置，将它们打包成了 view/controller/entity 三个包，如下：

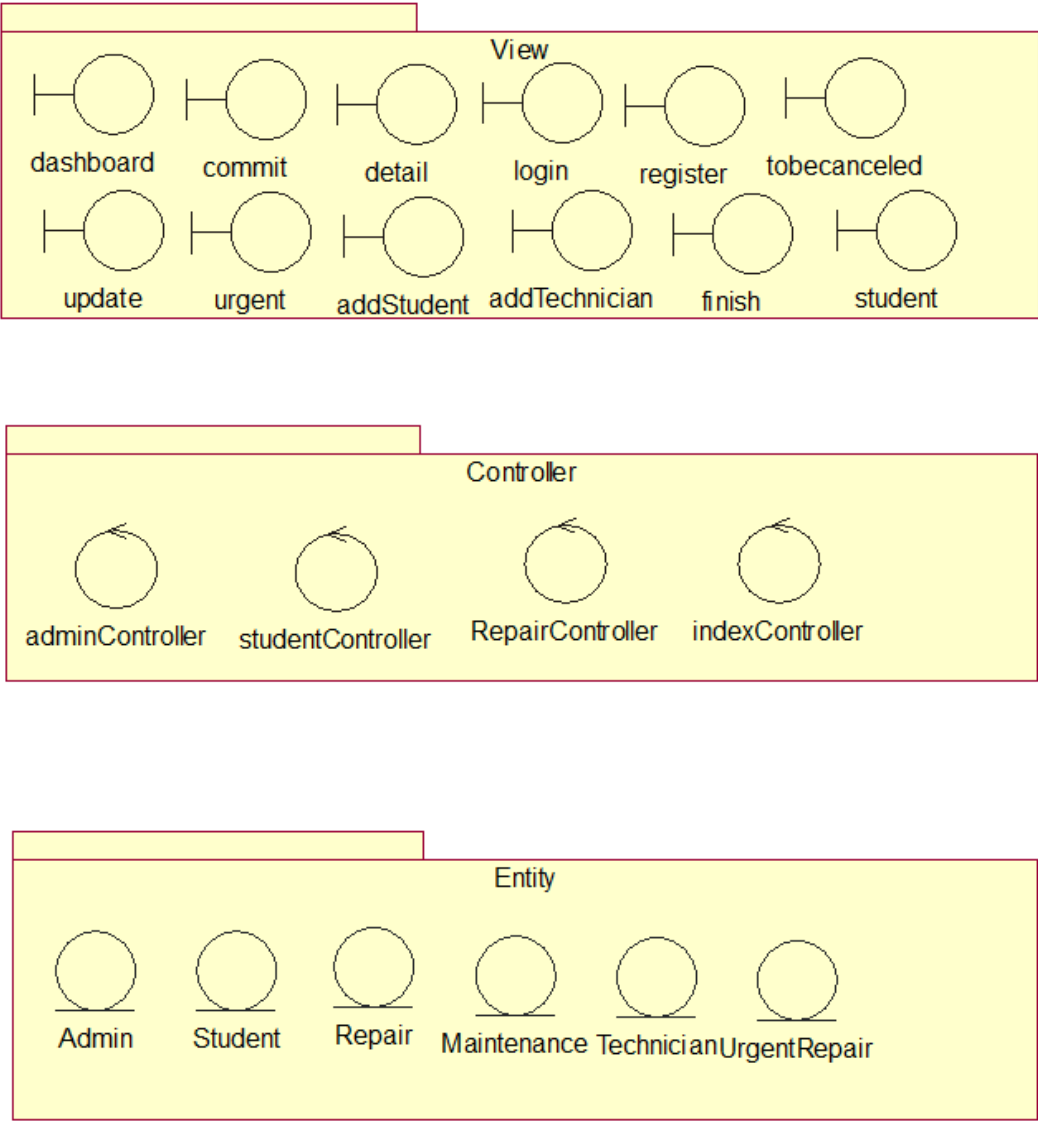


图 14：物业报修系统架构图

2.2 子系统及接口设计描述

首先，本项目之前析取的类（分析类）进行进一步设计，确定不存在非单一逻辑类。其次，根据类的相关性原则及本系统功能特点，将析取出的类进行成组，用包表示。其三，为每个子系统定义接口（Interface），并且复审。最后，将那个 Interface classes 与其相关的子系统建立连接关系。本项目中存在许多子系统，下面针对学生注册子系统/学生管理子系统/报修单管理子系统展开描述

2.2.1 学生注册子系统

边界：包括学生注册页面，注册结果界面。学生注册页面负责接收注册信息，注册结果界面负责显示注册的结果信息。

控制类：一方面接受来自边界的请求，结合实体提供的数据返回结果；另一方面，也负责对实体数据的操作，是整个子系统的中枢部分。

实体层：内部有学生注册的信息与注册的结果信息。

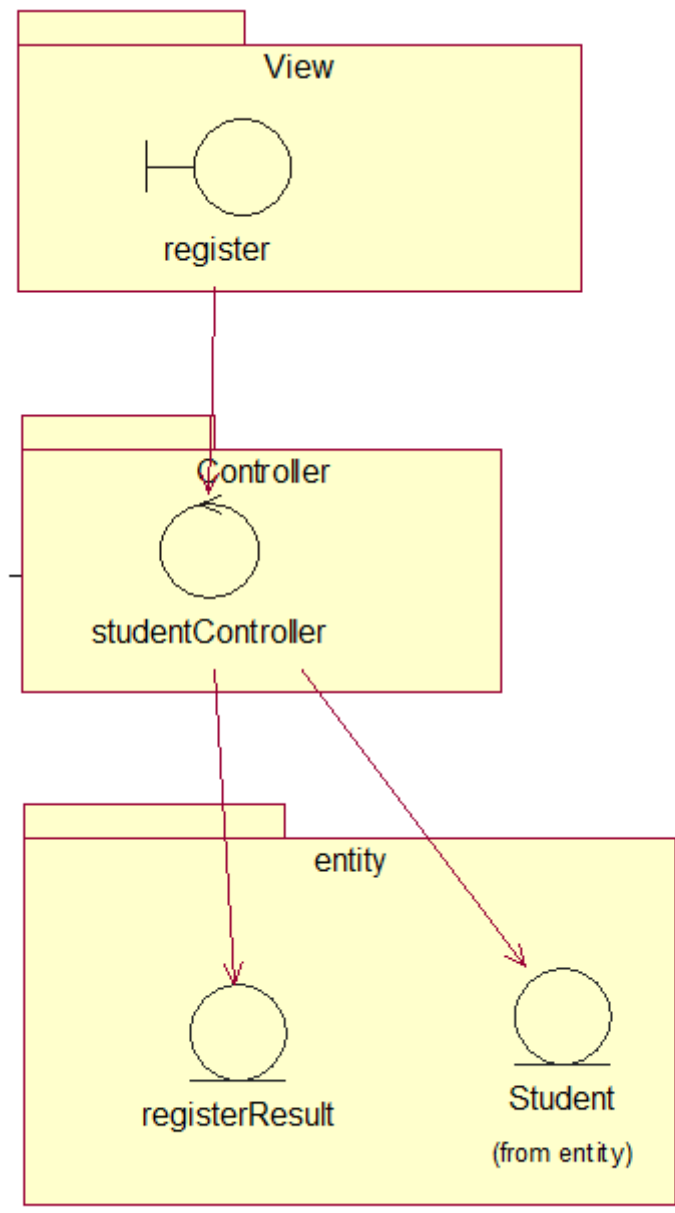


图 15：学生注册子系统架构图

2.2.2 学生管理子系统

边界：包括添加学生页面，查看学生界面。添加学生页面负责接收新学生信息，查看学生界面负责显示所有学生信息。

控制类：一方面接受来自边界的请求，结合实体提供的数据返回结果；另一方面，也负责对实体数据的操作，是整个子系统的中枢部分。

实体层：内部有学生信息。

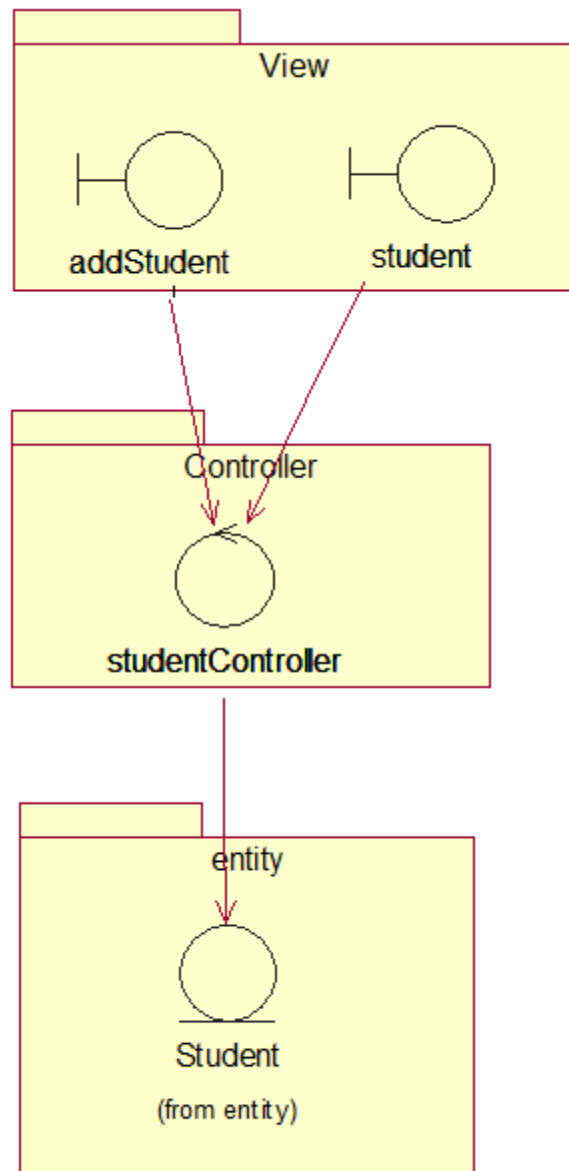


图 16：学生管理子系统架构图

2.2.3 报修单管理子系统

边界：包括报修单查看界面，提交报修单界面，更改报修单界面，取消报修单界面，验收报修单界面，报修单详情界面。

控制类：一方面接受来自边界的请求，结合实体提供的数据返回结果；另一方面，也负责对实体数据的操作，是整个子系统的中枢部分。

实体层：内部有报修单信息。

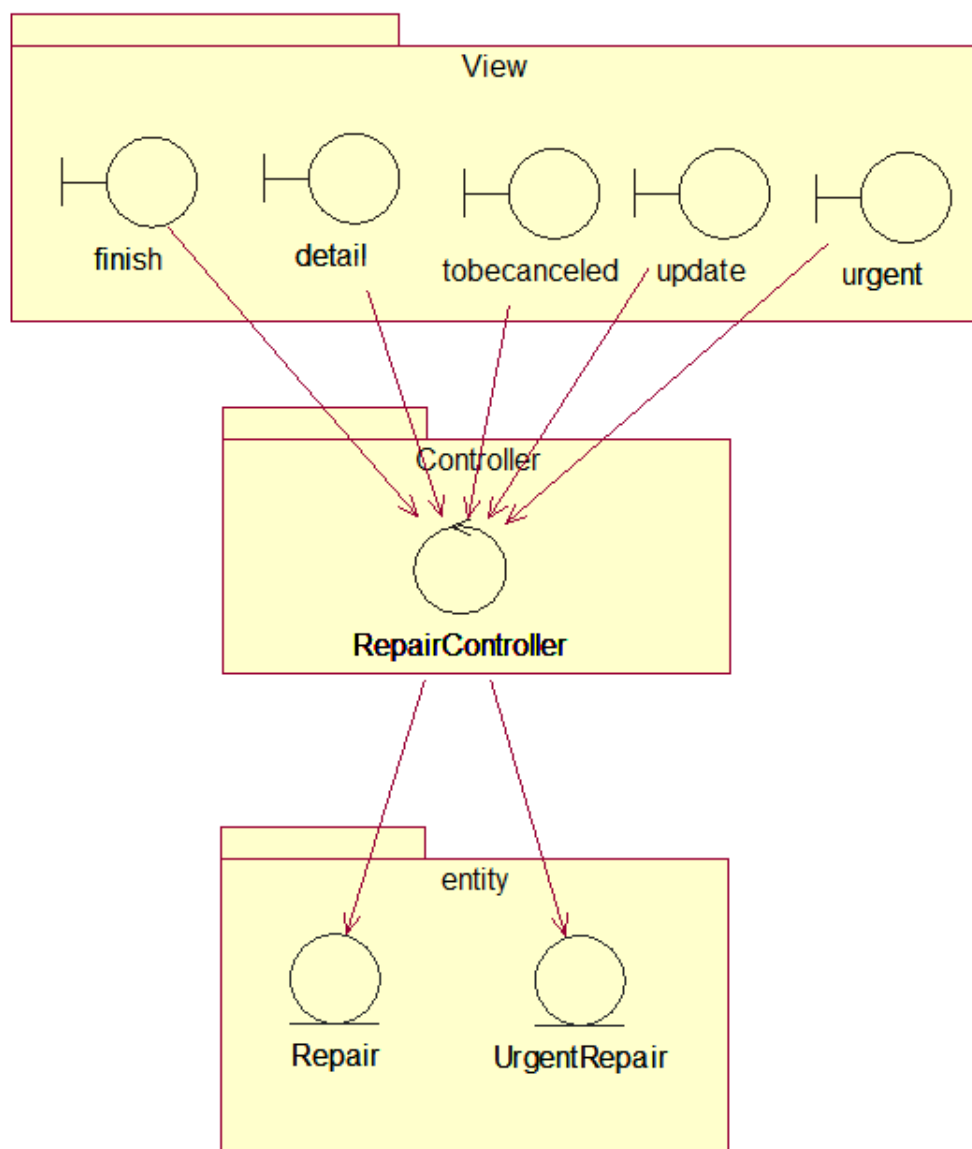


图 17：报修单管理子系统架构图

