

# 计算机科学系实验报告

课程名称	<u>UML 与可视化建模</u>	班级	<u>14 网络 2 班</u>		
实验名称	<u>电影购票平台</u>	指导教师	<u>曾少宁</u>		
姓名	<u>刘泽钊</u>	学号	<u>1414080903113</u>	日期	<u>2017.4.14</u>

---

## 一、实验目的

掌握基于 UML 2.0 的建模概念与方法，掌握各种 UML 图的概念与画法，其中包括用例图、活动图、类图、顺序图、组件图和状态图等。

## 二、实验设备与环境

操作系统：Windows 7；建模工具：StarUML。

## 四、实验要求

1. 实验及实验报告以增量方式完成，每次作业都在上一次作业的基础上完成，作业提交网站不提供报告下载，所以请同学们自行保管好自己的实验报告；
2. 请将实验报告中“占位符”信息替换为自己的实验相关信息；
3. 请认真撰写实验体会，实验课结束时立即上传实验报告：<http://zeng.shaoning.net/uml/>

## 四、实验内容、程序清单及运行结果

题目：电影购票平台

功能：1、购买电影票；

2、查询影视资讯。

### • 实验一：需求建模 - 用例模型

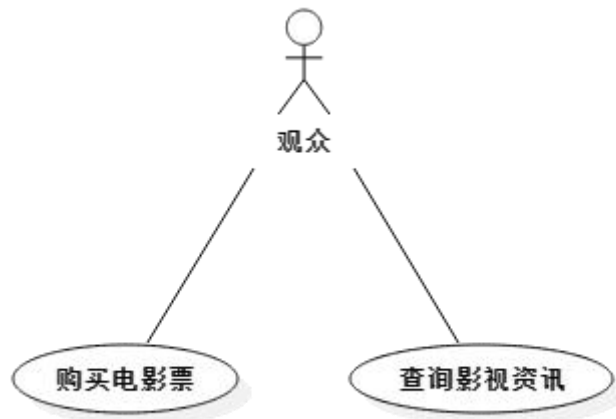


图 1：电影购票平台用例图

注：用例规约内容及项目可自行增加。

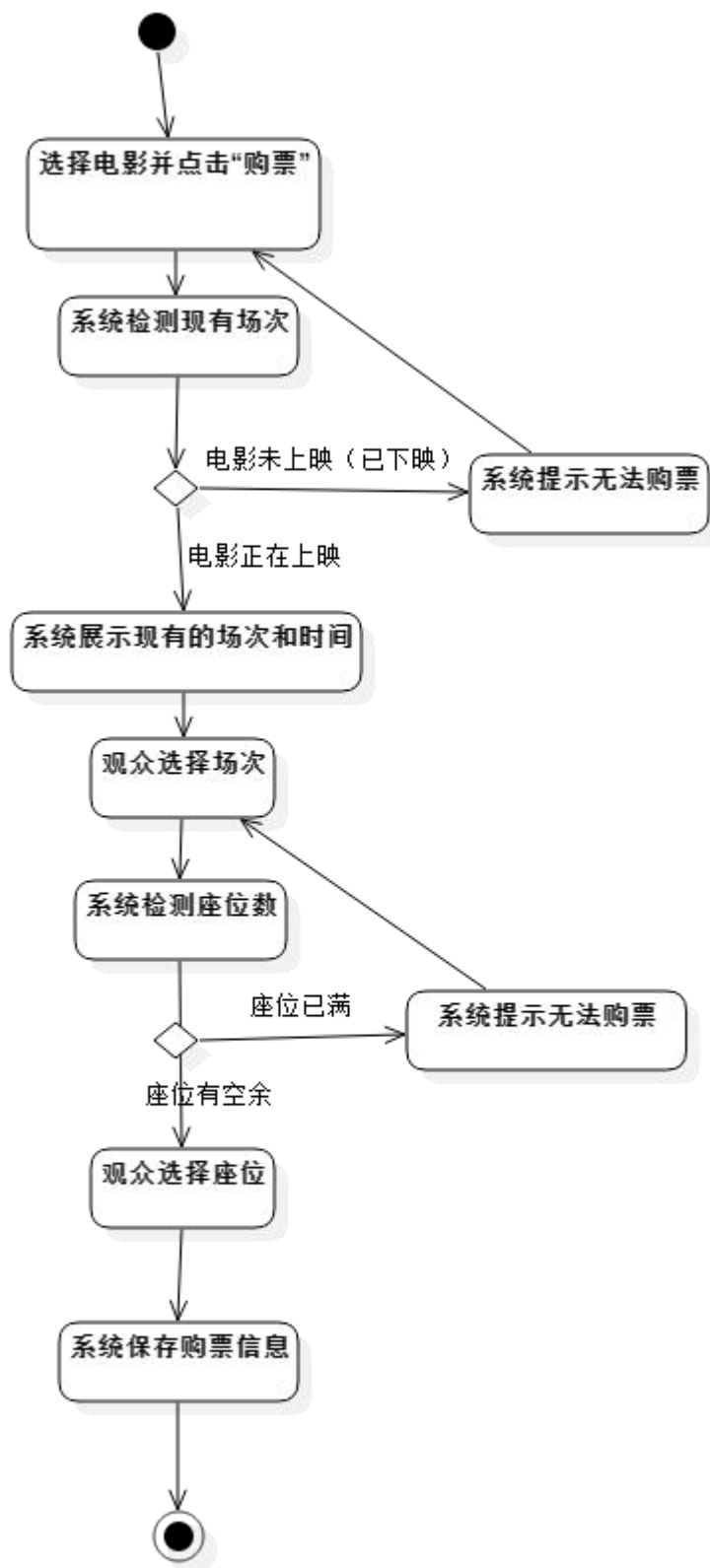
用例编号：	UC001
用例名称：	购买电影票
用例描述：	观众可在购票平台购买想要观看电影的电子票。
前置条件：	
基本流程：	<ol style="list-style-type: none"> <li>1、观众选择想要观看的电影并点击“购票”。</li> <li>2、系统检测场次，展示现有的场次和时间。</li> <li>3、观众选择想要观看的场次。</li> <li>4、系统检测座位数，显示可选的座位图。</li> <li>5、观众选择座位。</li> <li>6、系统保存购票信息。</li> </ol>
扩展流程：	<ol style="list-style-type: none"> <li>2.1 系统检测到电影未上映（已下映），提示观众不能购票。</li> <li>4.1 系统检测到该场次座位满，提示观众不能购票。</li> </ol>
后置条件：	

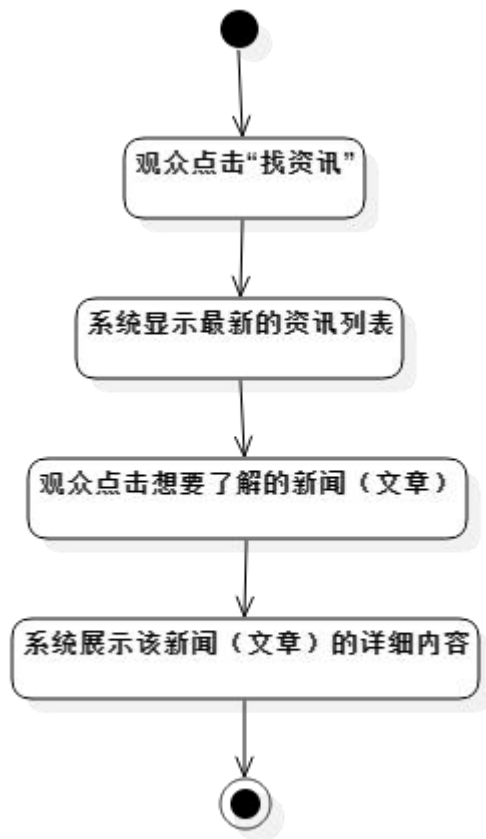
用例编号：	UC002
用例名称：	查询影视资讯
用例描述：	让观众获取各类影视的资讯。
前置条件：	
基本流程：	<ol style="list-style-type: none"> <li>1、观众点击“找资讯”。</li> <li>2、系统显示最新的资讯列表。</li> <li>3、观众点击想要了解的新闻（文章）。</li> <li>4、系统展示该新闻（文章）的详细内容。</li> </ol>
扩展流程：	
后置条件：	

• **实验二：过程建模 - 活动模型**

使用活动图描述系统的业务过程。

方法：将用例规约中的基本流程与扩展流程抽象为过程步骤（Action），画出对应的活动图。



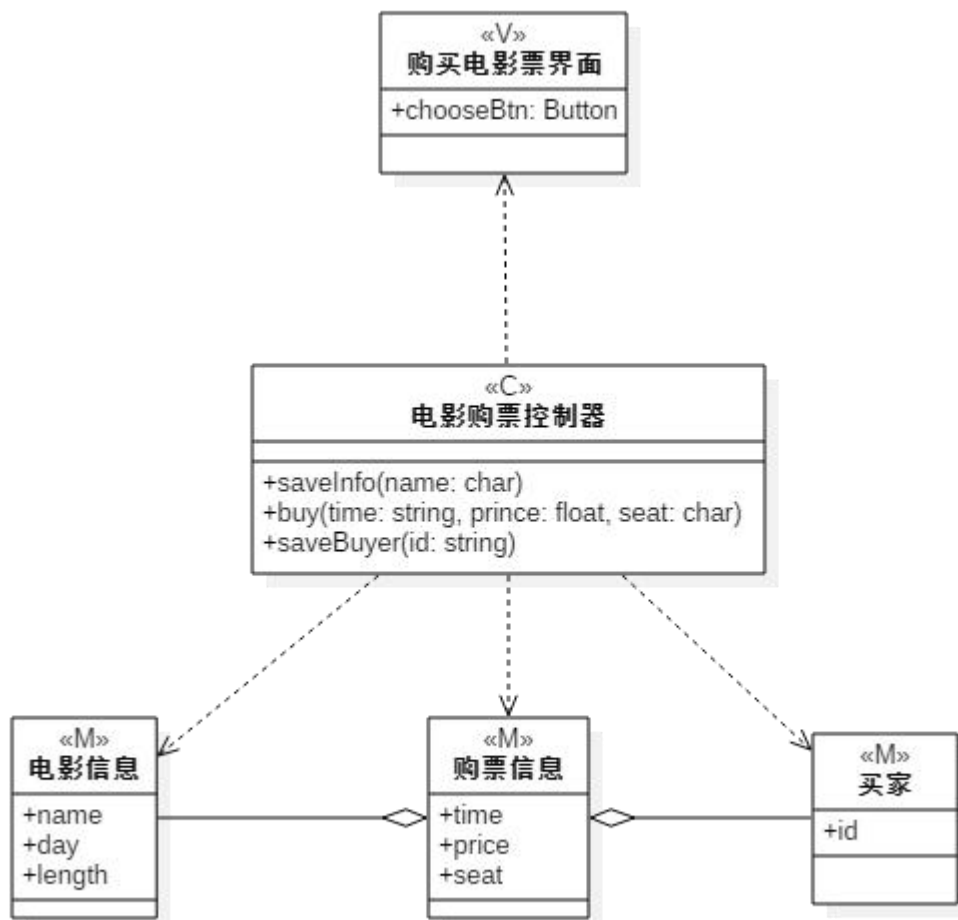


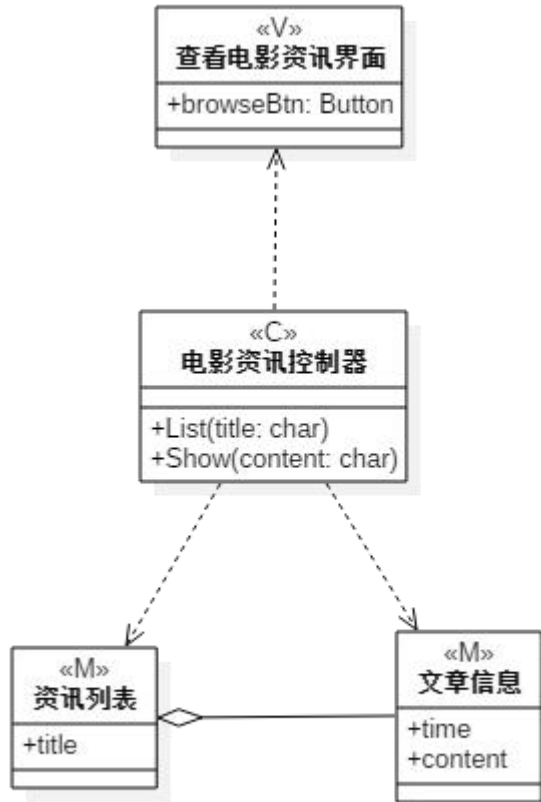
- **实验三：逻辑建模 - 类模型**

基于 MVC 设计模式找出实现用例的类。

方法：分别找出实现用例的模型（Model）、视图（View）和控制器（Controller）类，确定类之间的关系及其关键属性，画出类图。

参考：讲义 P26 页。



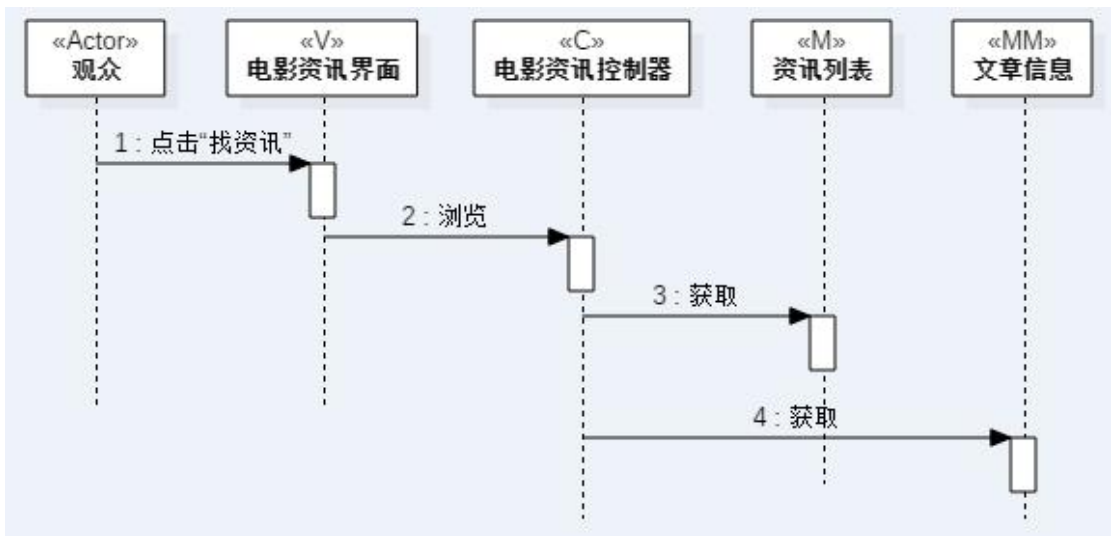
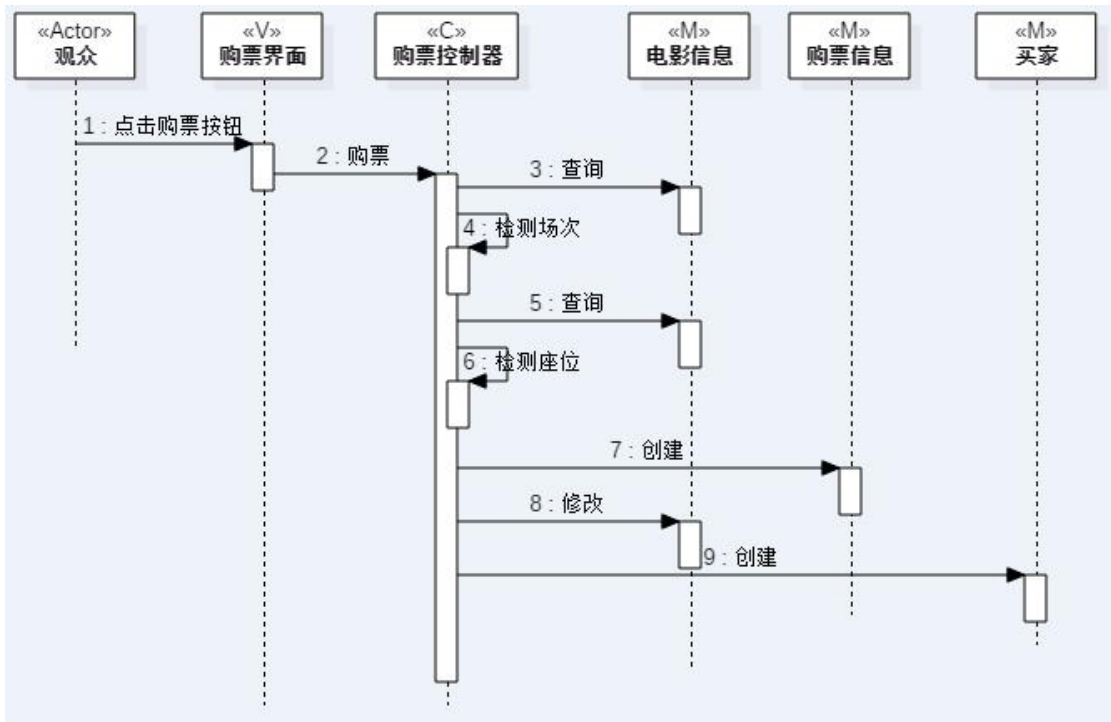


- **实验四：交互建模 - 顺序模型**

创建各个类（MVC 及 Actor）的对象，并描述对象之间的交互。

方法：分别创建参与者（Actor）、界面类（View）、控制器类（Controller）和模型类（Model）的对象，描述各个对象之间的消息及其顺序，画出顺序图。

参考：讲义 P33 页 8.7.2。



• 实验五：状态建模 - 状态模型

对系统中最重要的对象进行状态建模。

方法：选择一种对象，定义该对象的状态，描述状态之间的切换及条件，画出状态图。

参考：讲义 P9 和 P10 页。

## 五、实验体会

### 实验一：

首先是确定你的软件具有哪些主要功能（两到三个功能），基本流程重点关注观众和系统的交互，简单来说就是当观众做什么操作时，系统相应地做出什么反应。

### **实验二：**

用例规约中的每有一个流程在活动图中就有一个过程步骤（Action）与之对应，每有一个扩展流程就有一个决策（Decision）与之对应。值得注意的是，每个活动图都有且只有一个起点和一个终点，这个是不能缺少的。

### **实验三：**

首先确定模型（Model）、视图（View）和控制器（Controller）类，然后确定其属性，如用户界面中，用户通过按钮进行操作，这就需要添加“Button”这个属性；控制器中就要描述出系统会做出哪些动作去响应用户的操作；模型中包含一些字符、数字等等。

### **实验四：**

首先添加足够的 Lifeline，加上 Actor（参与者），以及所有的模型（Model）、视图（View）和控制器（Controller）类，每有一个就是一个 Lifeline；用例中的扩展流程在顺序图中相当于一个 Self Message，紧接着才是下一个 Message；顺序图中可以加入 Reply Message, 但为了简便和准确性，本实验中已经省略。

### **实验五：**