

# 软件工程实验报告----机票预订系统

学号：2011743

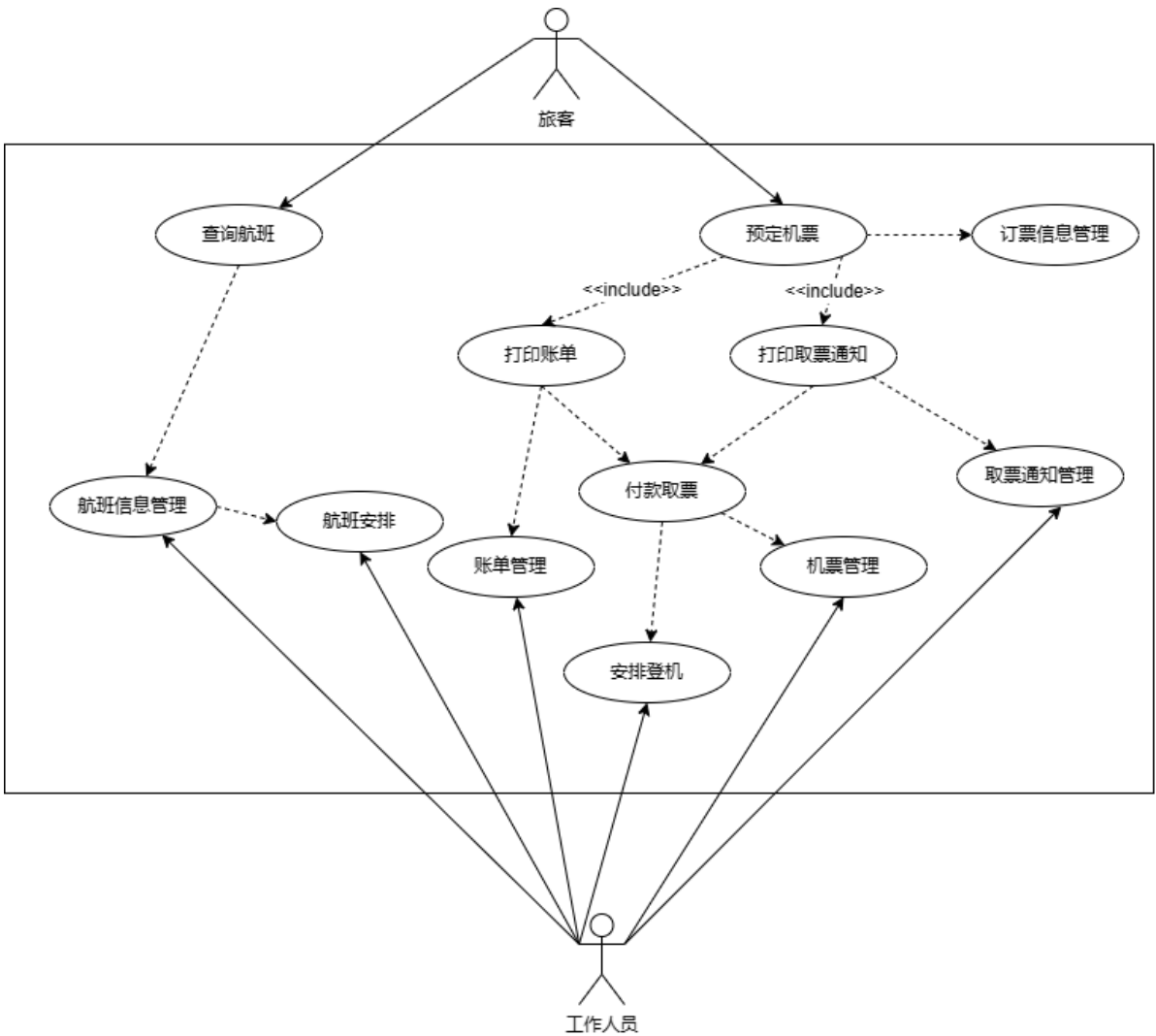
姓名：高祎珂

## 一、用例图

**定义：**由参与者（Actor）、用例（Use Case）以及它们之间的关系构成的用于描述系统功能的动态视图称为用例图。

**简要介绍：**用例图主要用于为系统的功能需求建模，它主要描述系统功能，也就是从外部用户的角度观察，系统应该完成哪些功能，有利于开发人员以一种可视化的方式理解系统的功能需求。同时也是为了方便用户和系统分析人员的理解尽量一致,为用户和系统分析人员提供一个大众的，一致性的方法,准确地把握系统的需求，能够节省时间，提高效率！

下面是我绘制的该机票预订系统的用例图：

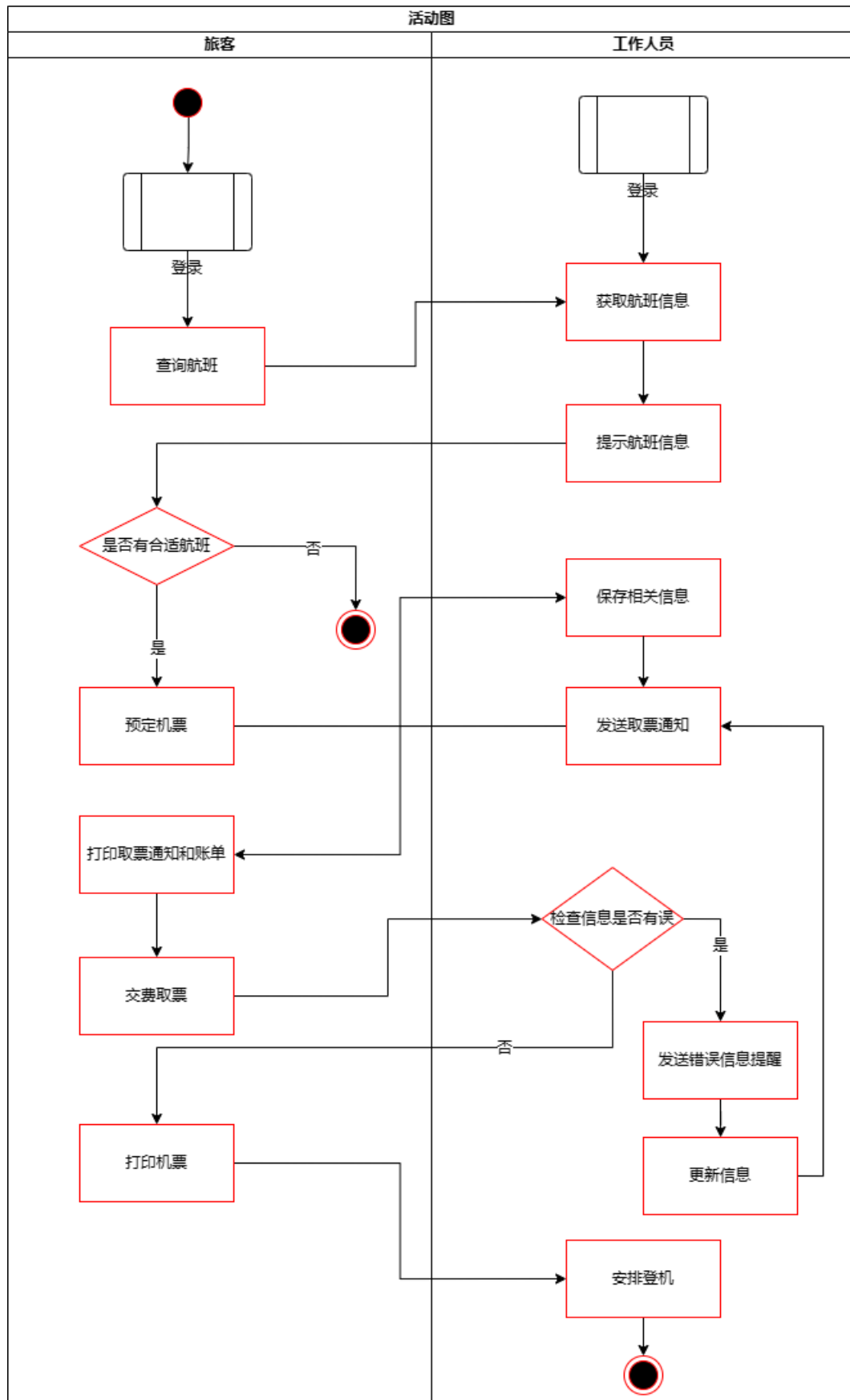


## 二、活动图

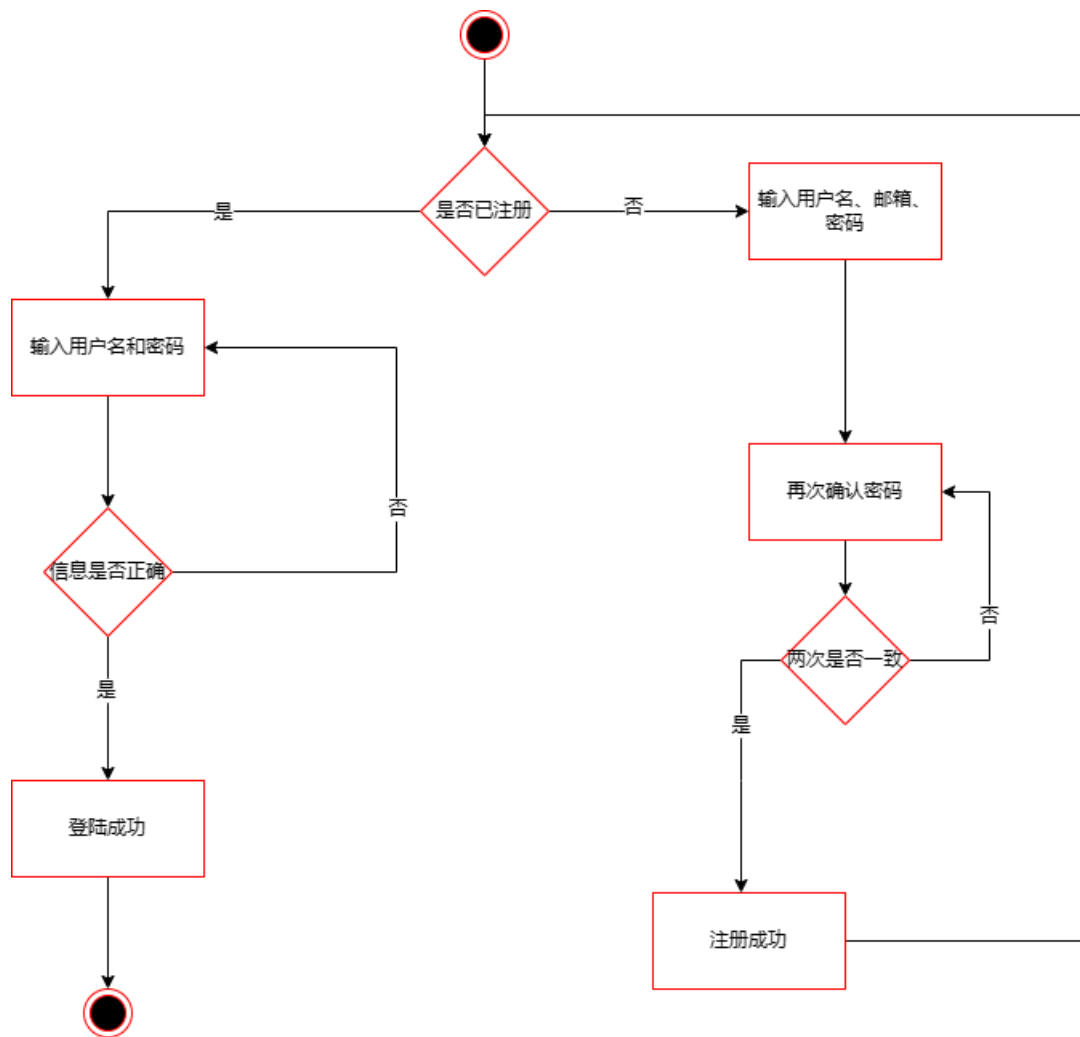
**定义：**阐明业务用例实现的工作流程。

**简要介绍：**活动图是UML用于对系统的动态行为建模的另一种常用工具，它描述活动的顺序，展现从一个活动到另一个活动的控制流。活动图在本质上是一种流程图。活动图着重表现从一个活动到另一个活动的控制流，是内部处理驱动的流程。

活动图描述的是对象活动的顺序关系所遵循的规则，它着重表现的是系统的行为，而非系统的处理过程。活动图能够表示并发活动的情形，活动图是面向对象的



活动图总图.jpg

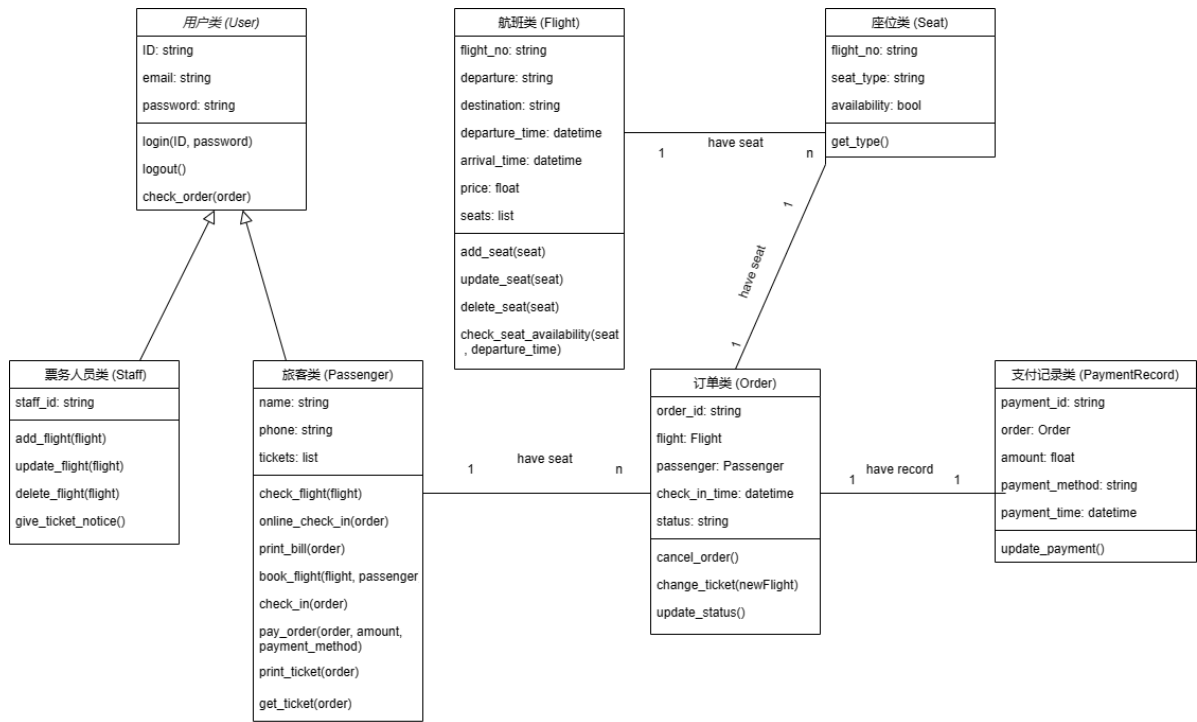


登陆子图.jpg

### 三、类图

**定义：**是由若干类关联在一起，反映系统或者子系统组成结构的静态图。

**简要介绍：**类图的建模贯穿工程的分析和设计阶段的始终，用来描述系统的静态部分。



## 四、顺序图

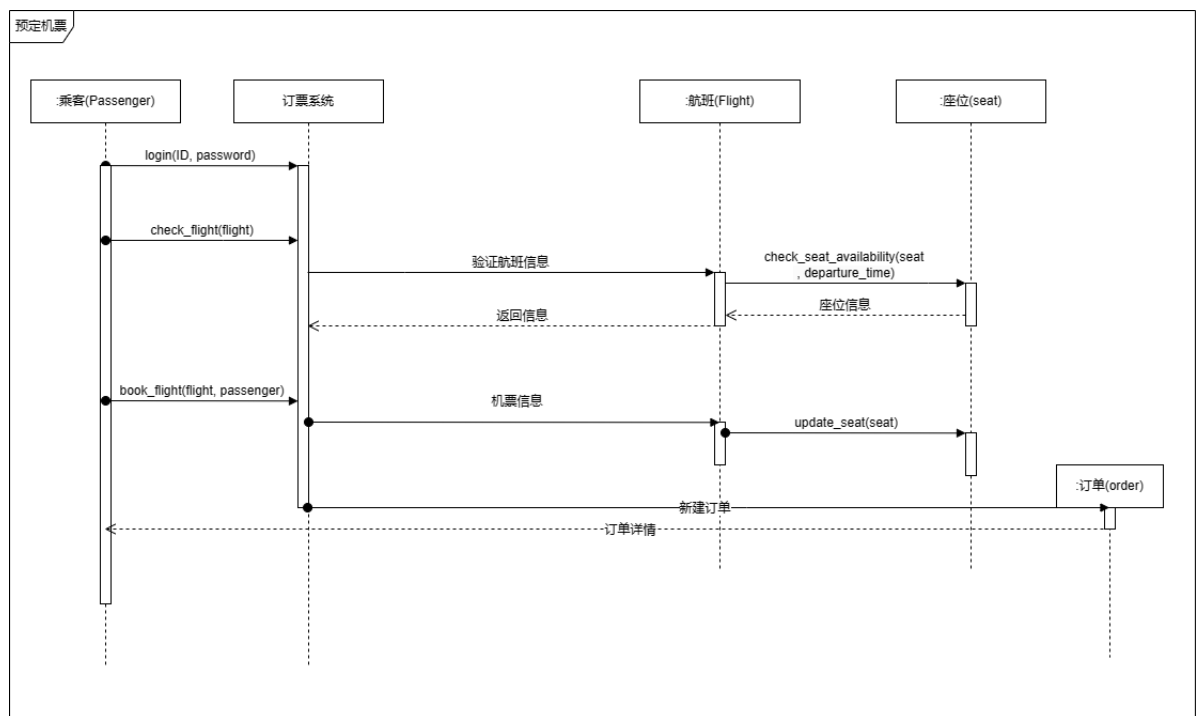
**定义：**是对对象之间传送消息的时间顺序的可视化表示。

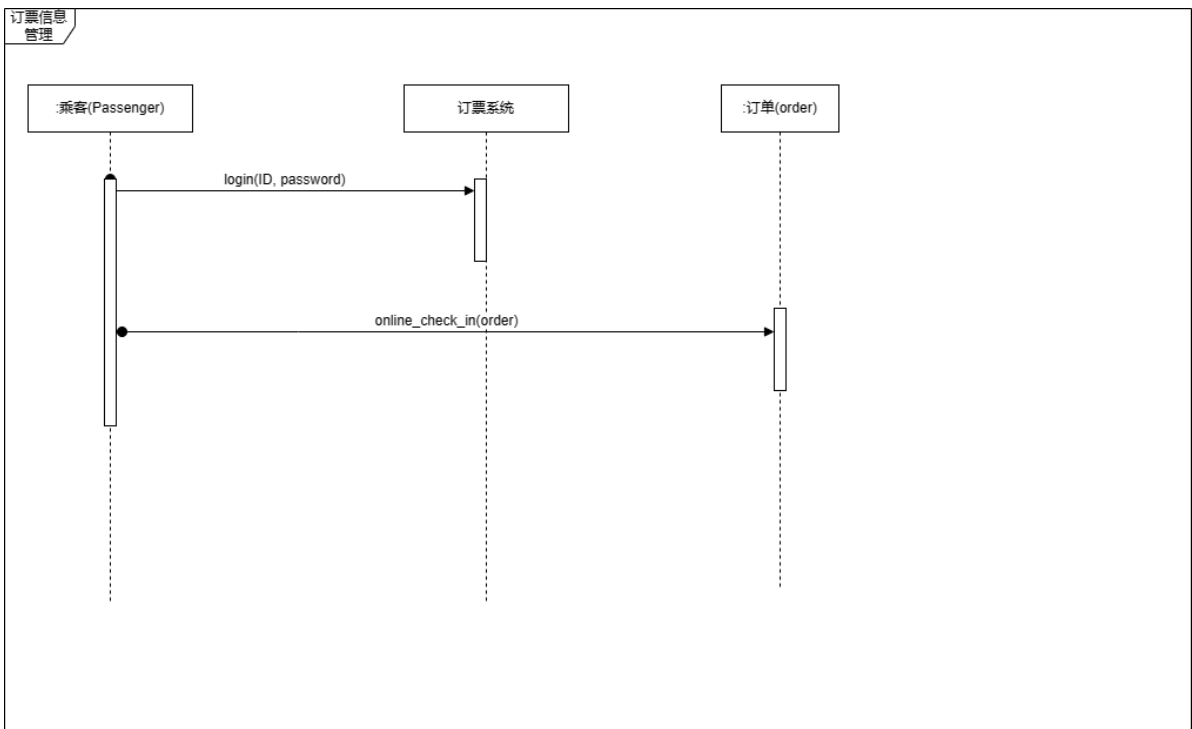
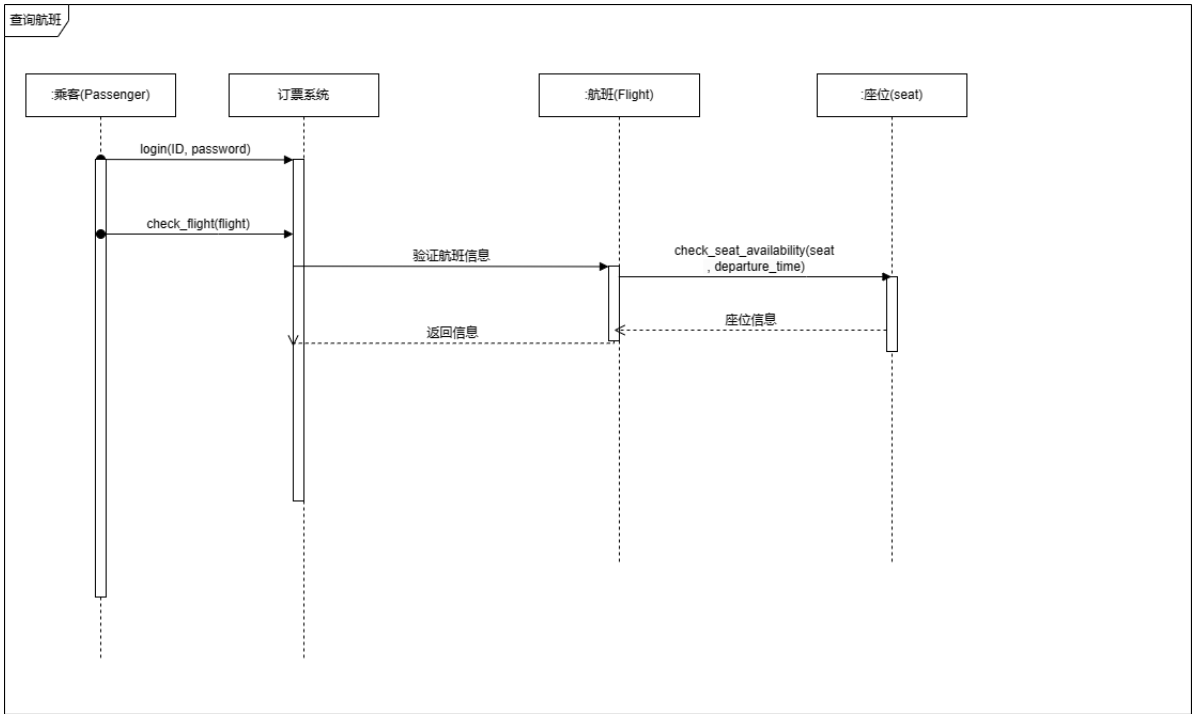
**简要介绍：**序列图的目的在于描述系统中各个对象按照时间的顺序的交互过程。

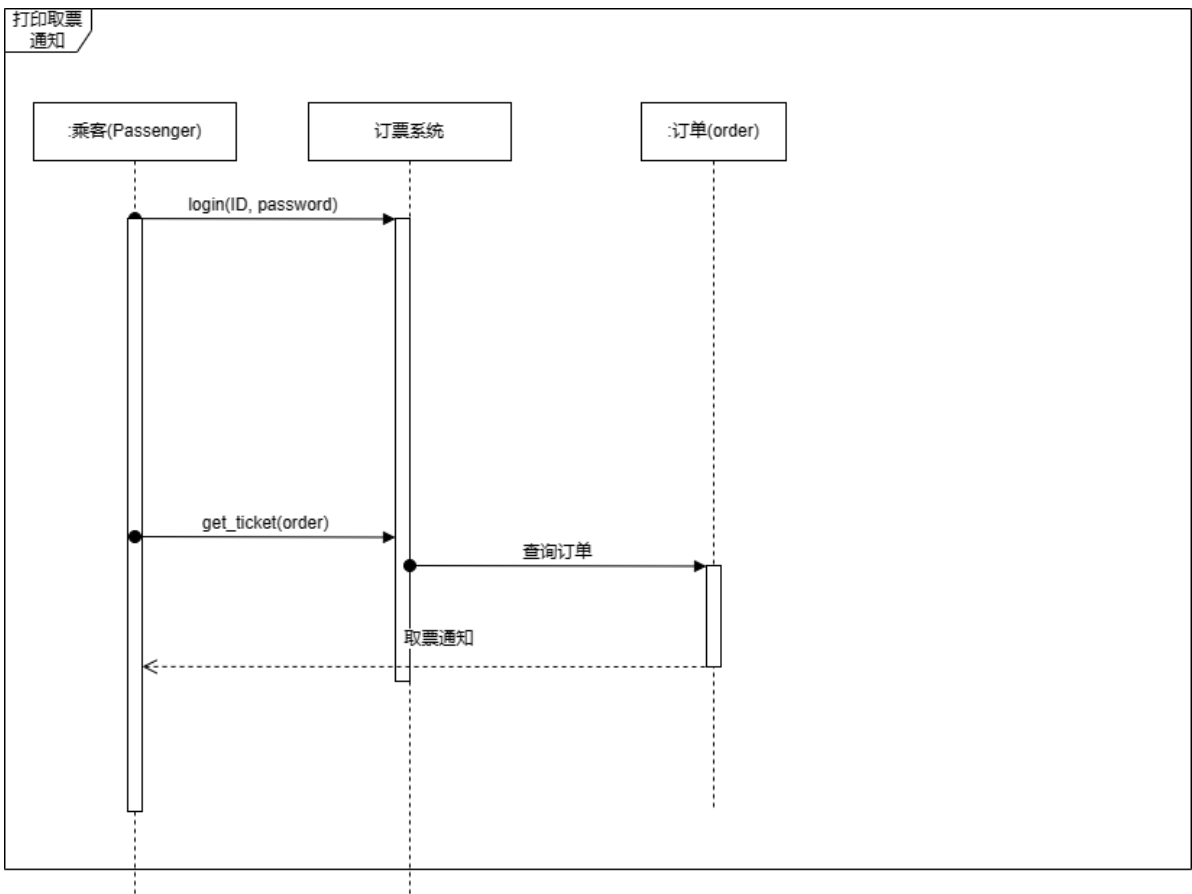
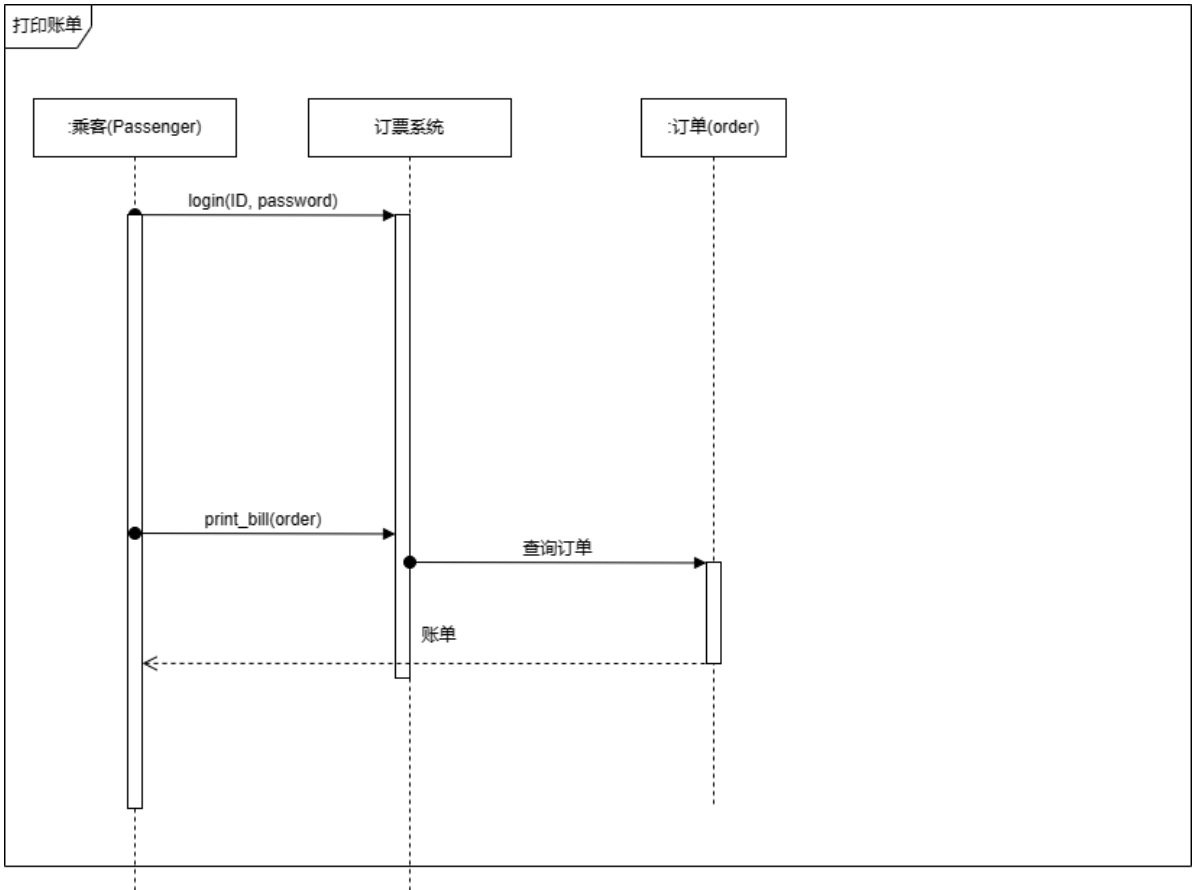
序列图将交互关系表示为一个二维图。纵向是时间轴，时间沿竖线向下延伸。横向轴代表了在协作中各独立对象的类元角色。类元角色用生命线表示。当对象存在时，角色用一条虚线表示，当对象的过程处于激活状态时，生命线是一个双道线。

消息用从一个对象的生命线到另一个对象生命线的箭头表示。箭头以时间顺序在图中从上到下排列。

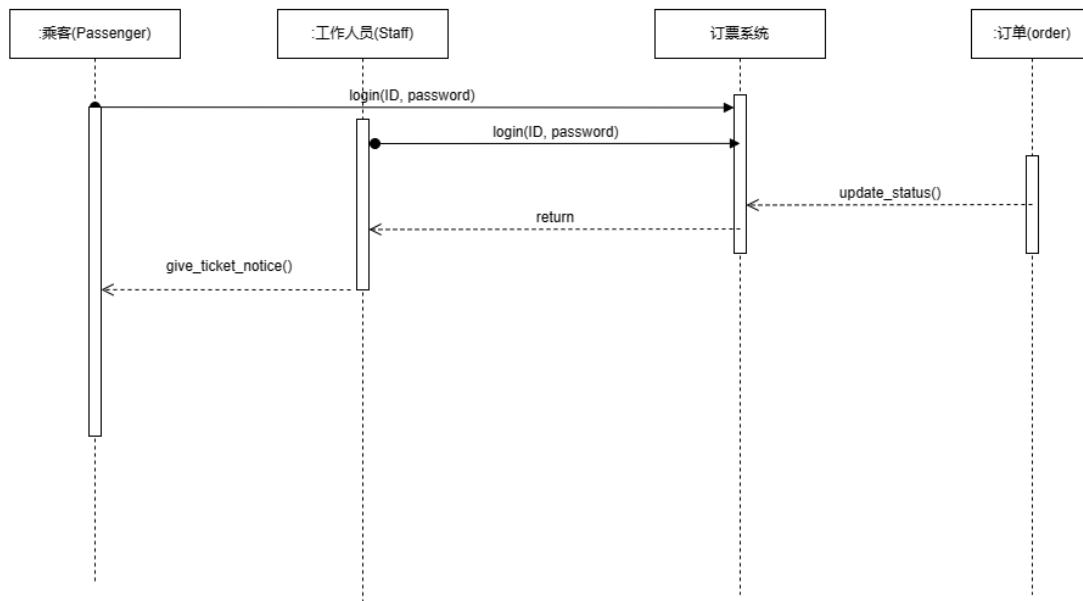
顺序图与用例图是一一对应的，下面是与12个用例对应的顺序图：



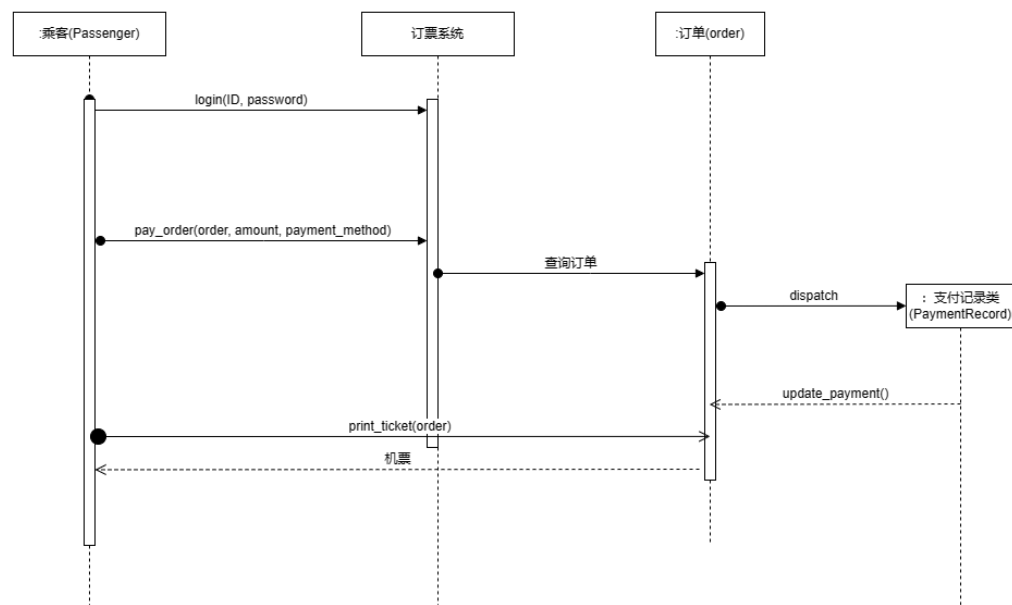




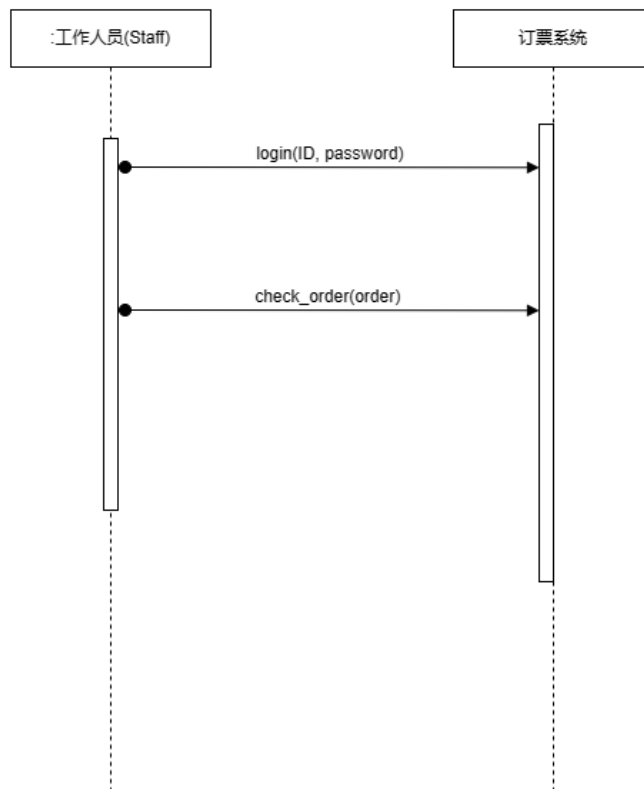
取票通知  
管理



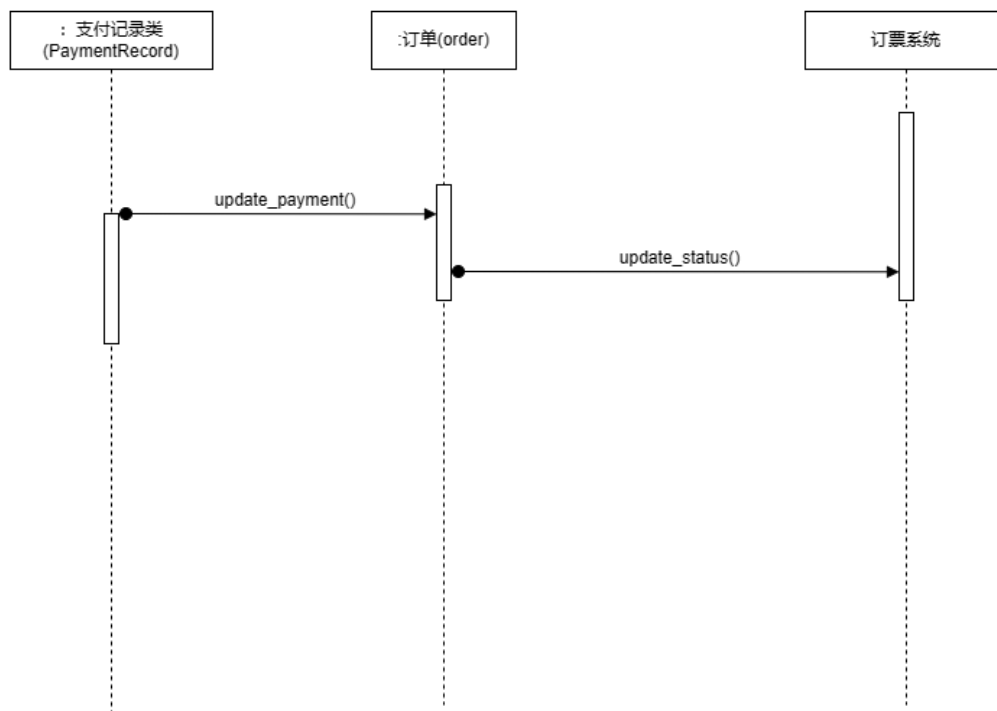
付款取票



机票管理

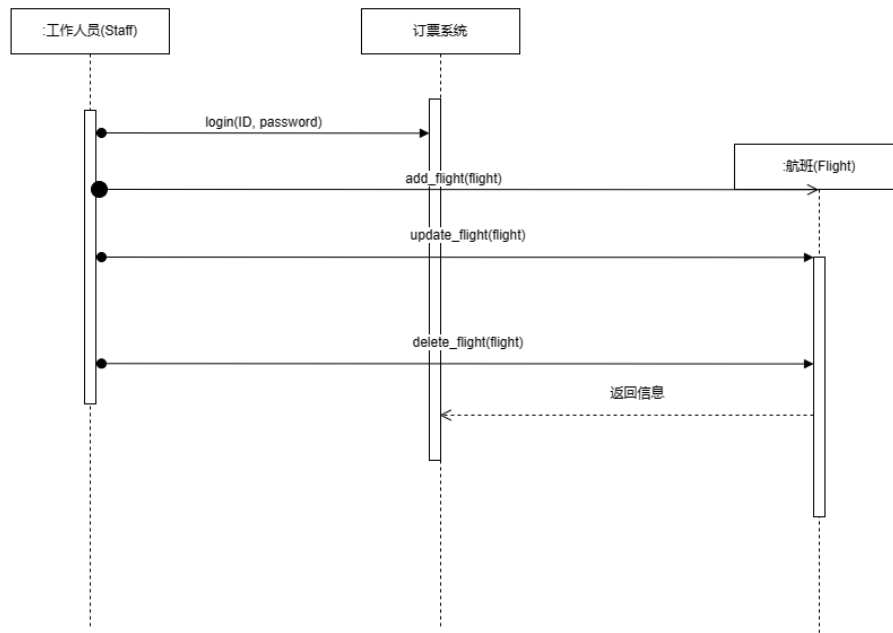
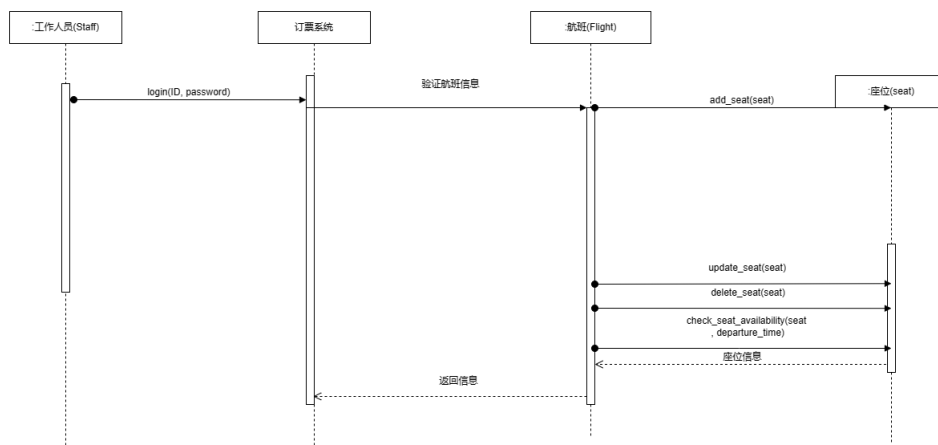


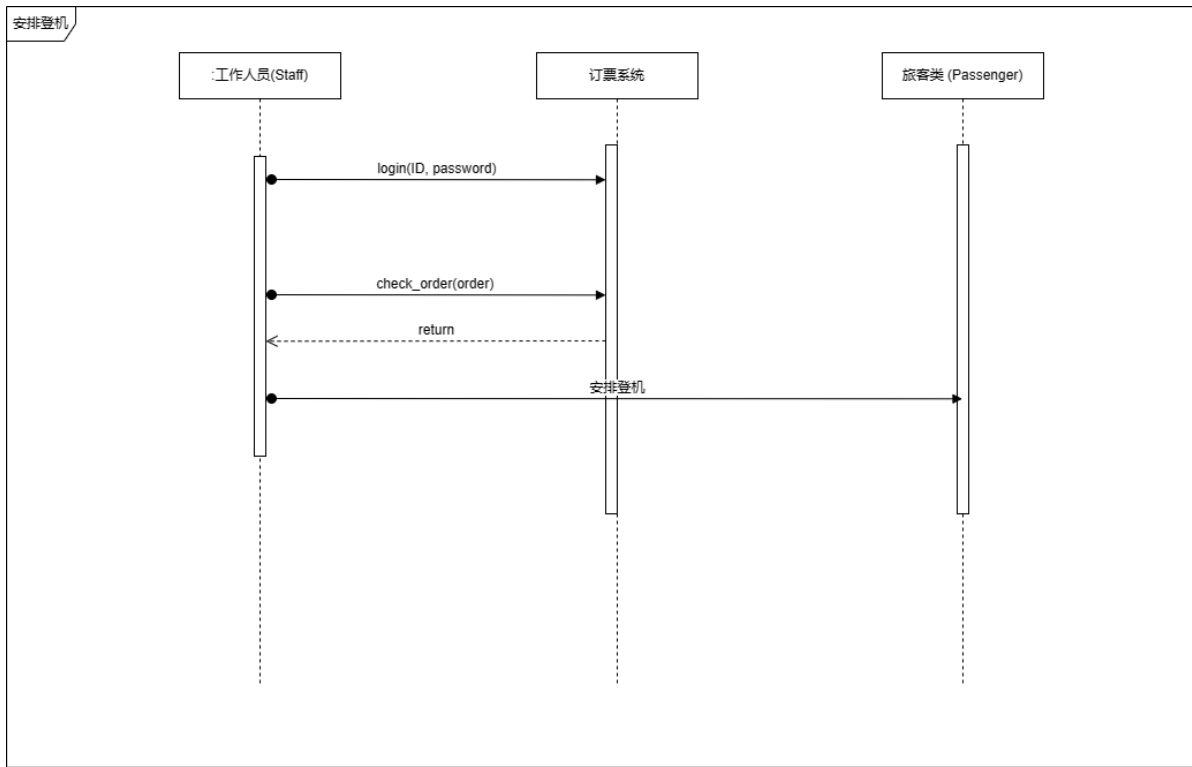
账单管理





## 航班安排

航班信息  
管理



## 五、协作图

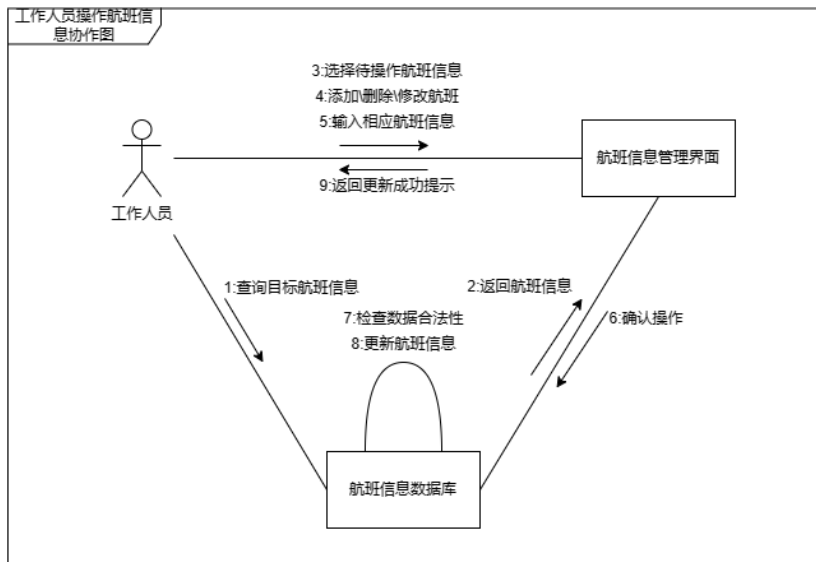
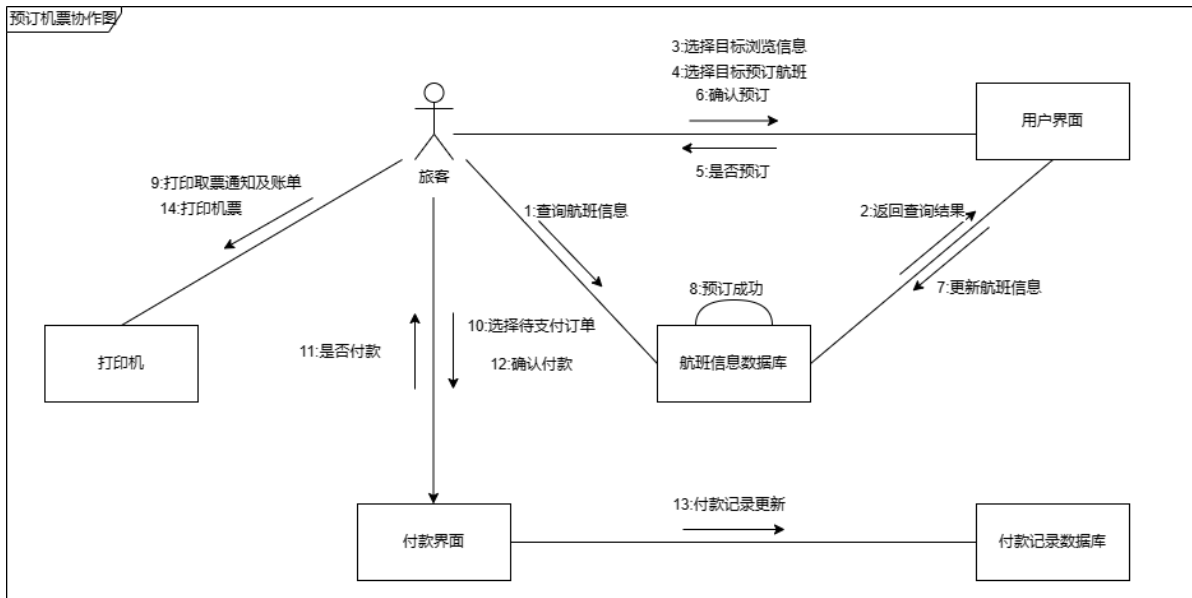
**定义：**描述相互合作对象间的交互关系。

**简要介绍：**UML协作图是一种交互图，强调的是发送和接收消息的对象之间的组织结构。

UML协作图用于显示对象之间如何进行交互以执行特定用例或用例中特定部分的行为。设计员使用UML协作图和序列图确定并阐明对象的角色，这些对象执行用例的特定事件流。它们是主要的信息来源，用于确定类的职责和接口。

与序列图不同，UML协作图显示了对象之间的关系。序列图和UML协作图表述的是相似的信息，但表述的方式却不同。UML协作图显示对象之间的关系，它更有利于理解对给定对象的所有影响，也更适合过程设计。

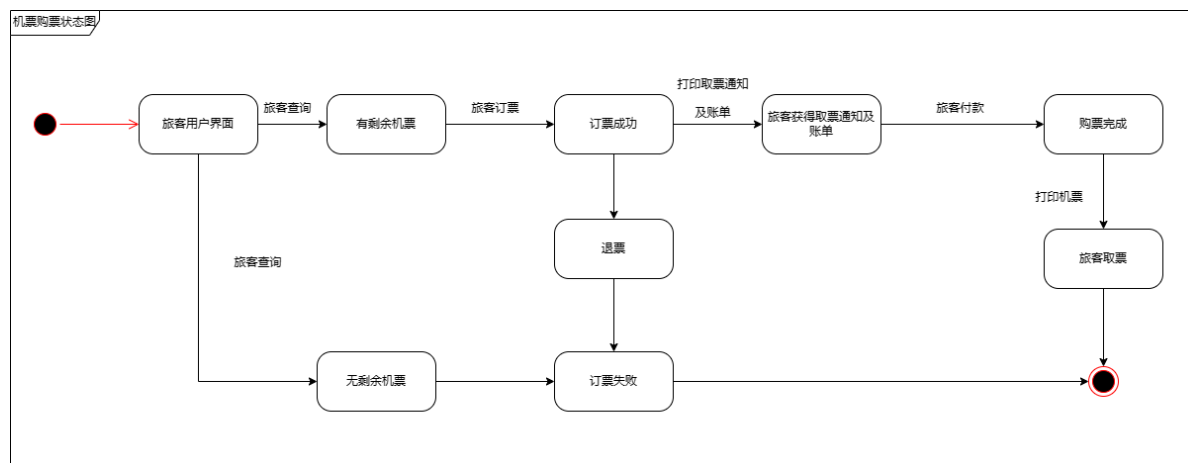
UML协作图的格式决定了它们更适合在分析活动中使用。它们特别适合用来描述少量对象之间的简单交互。随着对象和消息数量的增多，理解UML协作图将越来越困难。此外，UML协作图很难显示补充的说明性信息，例如时间、判定点或其他非结构化的信息，而在序列图中这些信息可以方便地添加到注释中。



## 六、状态图

**定义：**用来描述一个特定对象的所有可能状态以及由于各种事件的发生而引起的状态之间的转移。

**简要介绍：**状态图(Statechart Diagram)是描述一个实体基于事件反应的动态行为，显示了该实体如何根据当前所处的状态对不同的事件做出反应的。通常我们创建一个UML状态图是为了以下的研究目的：研究类、角色、子系统、或组件的复杂行为。一般可以用状态机对一个对象的生命周期建模，状态图用于显示状态机（State Machine Diagram），重点在与描述状态图的控制流。

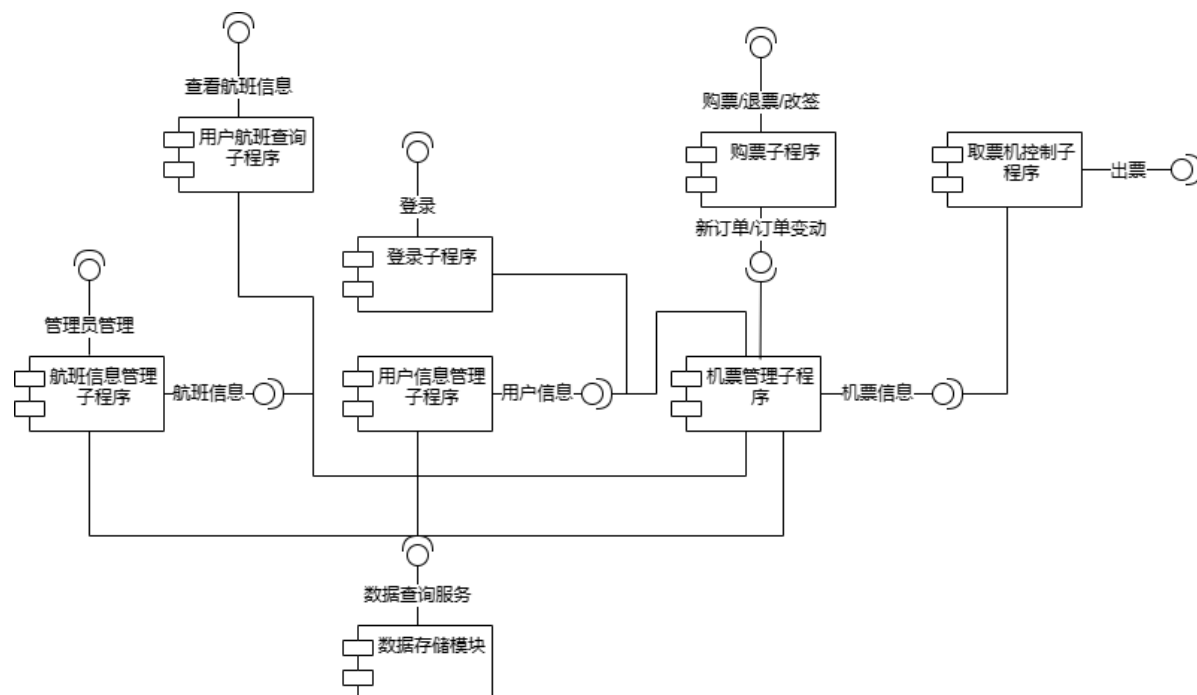


## 七、构件图

**定义：**主要用于描述各种软件构件之间的依赖关系。

**简要介绍:** 构件图是从软件架构的角度来描述一个系统的主要功能，如子系统、类、包、构件等。

构件是系统中遵从同一组接口且提供其实现的物理的、可替换的部分。使用构件最重要的是复用。构件图提供系统物理视图，在一个非常高的层次上显示系统中的构件与构件之间的依赖关系。



## 八、部署图

**定义：**描述了一个系统运行时的硬件节点，在这些节点上运行的软件构件将在何处物理运行以及它们将如何彼此通信的静态视图

**简要介绍：**系统开发人员和部署人员可以利用部署图去了解系统的物理运行情况。如果，开发的软件系统只需在一台计算机上运行，且使用的标准设备，则不需要为它画出系统部署图。部署图只需要给那些复杂的物理运行情况进行建模。

部署图显示了系统的硬件，安装在硬件上的软件，用于连接硬件的各种协议和中间件等。

部署模型的目的：描述一个具体应用的主要部署结构，通过对各种硬件，在硬件中的软件以及各种连接协议的显示，可以很好的描述系统是如何部署的；平衡系统运行时的计算资源分布；可以通过连接描述组织的硬件网络结构或者是嵌入式系统等具有多种硬件和软件相关的系统运行模型。

