

Министерство образования и науки Российской Федерации  
федеральное государственное автономное образовательное  
учреждение высшего образования  
«Санкт-Петербургский национальный исследовательский  
университет информационных технологий, механики и оптики»  
(Университет ИТМО)

**Факультет:** Инфокоммуникационных технологий

**Направление (специальность)** 11.04.02 «Инфокоммуникационные  
технологии и системы связи»

## **ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 4**

По дисциплине:

«Проектирование информационных систем»

На тему:

«Проектирование объектной модели»

**Выполнил:**

студент(ка) гр. К4113с

Логинова В. С.

**Преподаватель:**

Осипов Н. А.

Санкт-Петербург

2020

**Цель работы:** изучить основы разработки объектных моделей с использованием шаблонов GRASP для распределения обязанностей между классами.

**Задачи:**

Для выбранного варианта инфокоммуникационной системы выполнить следующее:

На основе модели предметной области (основные сущности в терминах предметной области задачи) разработать объектную модель (диаграмма последовательностей) и диаграмму классов (UML), применяя для распределения обязанностей между классами принципы объектного проектирования (шаблоны GRASP).

**Программное обеспечение:** draw.io

**Ход работы:**

С использованием принципов объектного проектирования была создана диаграмма классов (рисунок 1), включающая в себя ряд сущностей и наследуемых связей. Она требуется для формализации распределения обязанностей между классами системы и обозначения, что входит в конкретный модуль системы.

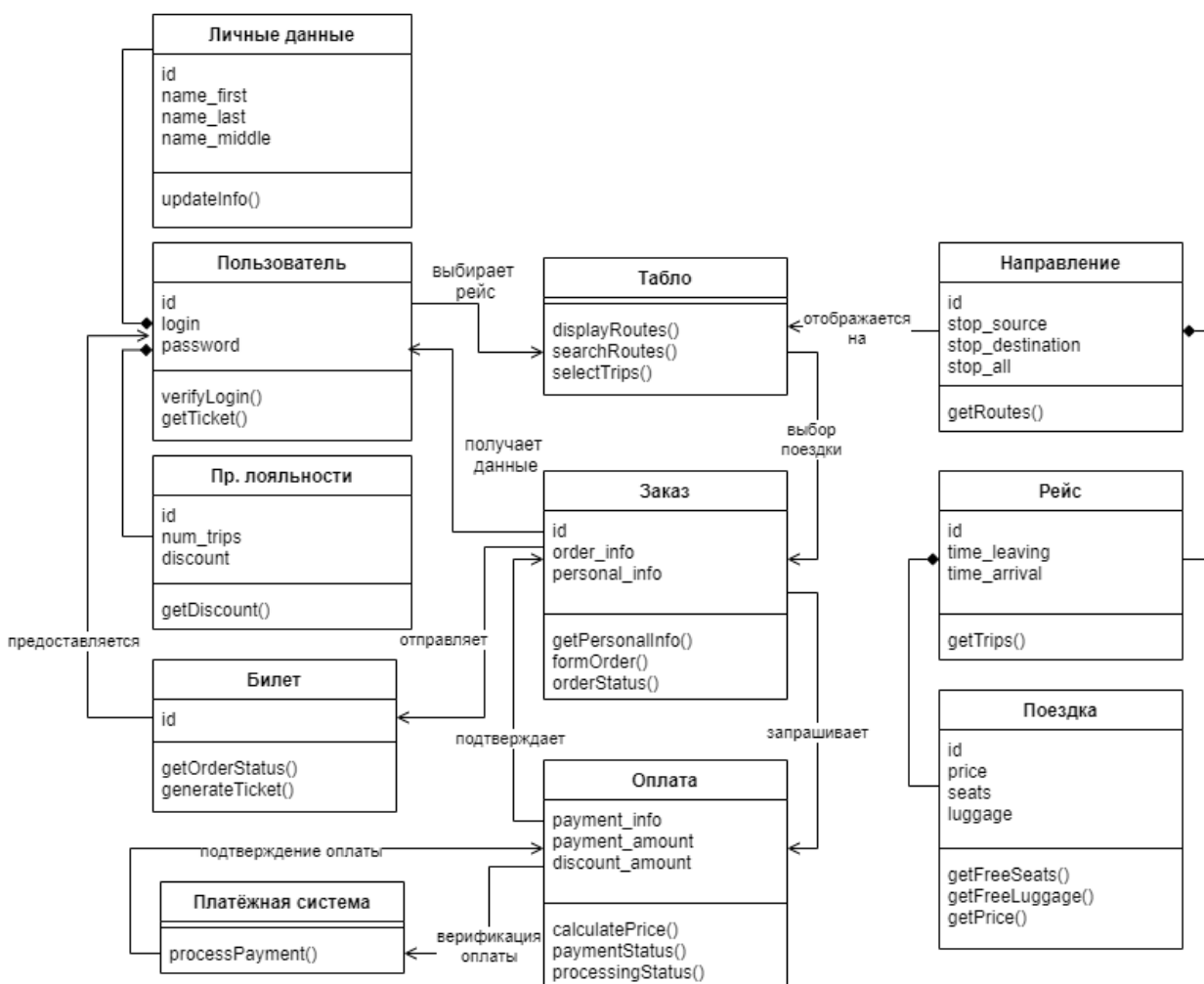


Рисунок 1 – Диаграмма классов

Применённые при проектировании диаграммы классов шаблоны GRASP:

### 1. Information Expert & Indirection

Класс «Табло» выполняет роль информационного эксперта, агрегируя информацию о направлениях перевозок. Таким образом, он обладает наибольшей полнотой информации, которая в дальнейшем используется пользователем для совершения действий с системой.

В представленной диаграмме «Табло» - один из ключевых корневых объектов. Он выступает также как промежуточный объект (view) для данных о маршрутах и рейсах, обеспечивая таким образом слабое зацепление между объектами и возможность их повторного использования.

### 2. Controller

Классы «Табло» и «Заказ» отвечают на операции, запросы на которые приходят от пользователя. Они могут выполнять сценарии вариантов использования, и выступают как обработчики информации, делегируя задачи вроде оплаты и генерации билета компетентным исполнителям (соответствующим классам).

### 3. Low Coupling & High Cohesion

Посредством создания ряда отдельных классов и установления между ними взаимосвязей наследования данных и присвоения отдельных функций смогли установить относительно слабое зацепление и высокую связность компонентов.

Вторая диаграмма, созданная в ходе работы – диаграмма последовательностей, показывающая взаимодействие объектов, упорядоченное по времени (рисунок 2).

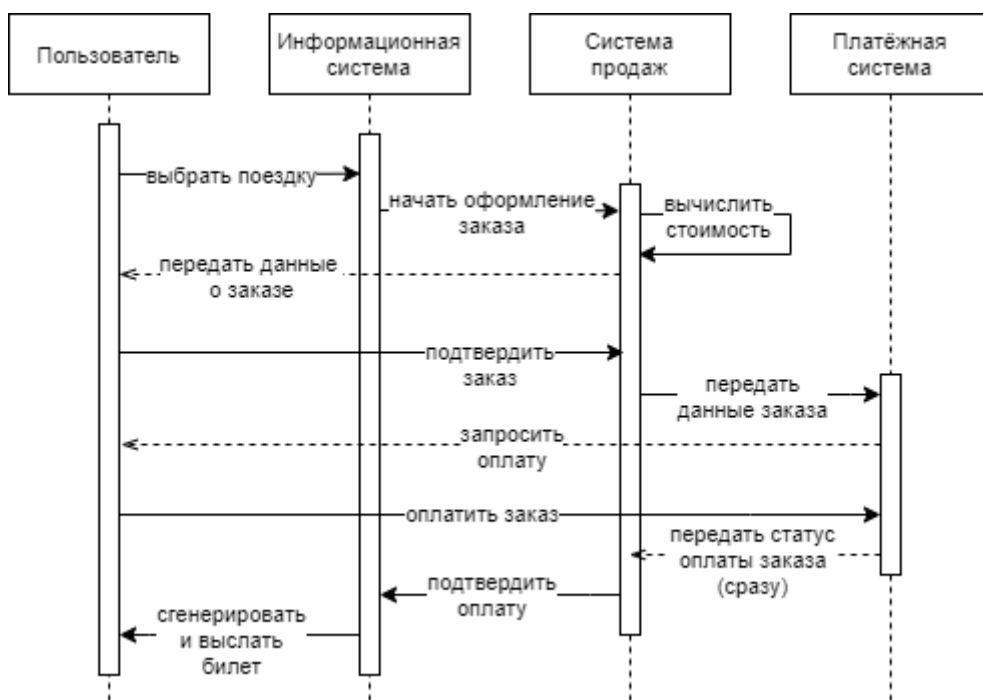


Рисунок 2 – Диаграмма последовательностей

Аналогично диаграмме классов, для диаграммы последовательностей применили ряд шаблонов GRASP для объектно-ориентированного программирования, структурировав всю систему в 3 подсистемы – информационную, продаж и платёжную. Большая часть процессов выполняется системой продаж – Information Expert для всех данных о рейсах и их стоимости. Информационная система выступает в роли view для этих данных, и с ней происходит основное взаимодействие пользователя. Подсистемы обладают свойствами низкого зацепления и высокой связности – все основные функции выполняются отдельно.

### **Выводы:**

В ходе работы были построены диаграмма классов и диаграмма последовательностей системы продажи билетов на междугородние маршруты.

Благодаря выполненной работе достигнуто понимание того, каким образом структурированы классы, входящие в систему, как между ними построено взаимодействие. Были изучены и применены шаблоны GRASP, позволяющие строить системы с распределением ответственностей в объектно-ориентированном программировании.