***Штуль Ксении***

***гр. ИСТ-107***

**Диаграмма видов деятельности**

Диаграмма видов деятельности отражает динамику системы и особенно полезна при описании поведения, включающего в себя большое количество параллельных процессов, а также для моделирования поведения системы в самом общем виде на этапе анализа.

Поведение ПС «Туристическое агентство» можно представить с помощью диаграммы видов деятельности (рис. 1).

После авторизации, в зависимости от типа пользователя, открывается соответствующее меню. Клиент имеет возможность просмотреть список своих заказов. Администратор может добавить и изменить информацию о пользователях системы. Администратор может работать с клиентом: добавлять/редактировать клиента, редактировать проводимые акции, редактировать категории туров, добавлять/изменять заказы. После выполнения каждой операции пользователь видит сообщение о результате работы системы.

*Рис. 1 Диаграмма видов деятельности*

**Диаграмма состояний**

Диаграмма состояний отражает внутренние состояния объекта в течение его жизненного цикла от момента создания до разрушения. Обычно диаграммы состояний строятся для единственного класса, чтобы показать динамику поведения единственного объекта.

Диаграмма состояний для объекта класса «Заказ»:

****

*Рис. 2 Диаграмма состояний для объекта класса «Заказ»*

Из начального состояния объект переходит в состояние *«CreateOrder».* Далее наступает состояние *«LoadInformation»* или *«AddInformation»*, в зависимости от того, создан ли заказ ранее, или необходимо создать новый заказ. Из состояния *«AddInformation»* объект может перейти в состояние редактирования *«EditInformation»*или *«SaveChanges»* для сохранения изменений. Из состояния объект может также перейти в состояние *«EditStatus»* для изменения статуса заказа. Жизненный цикл объекта завершается состоянием *«SaveChanges»,* на котором производится сохранение объекта в БД.

**Диаграмма классов на стадии реализации**



*Рис. 3 Диаграмма классов на стадии реализации*

В данной программной системе можно выделить следующие классы:

1. *User* – класс, описывающий зарегистрированных в системе сотрудников, которые работают с системой. Класс имеет следующие атрибуты: *login* (имя пользователя), *password* (пароль), *fio* (ФИО). Класс находится в отношении с классом *UserRole*.
2. *UsersRole* – класс, описывающий роль пользователя в системе. Класс имеет следующие атрибуты: *role* (роль).
3. *Client* – класс, описывающий клиентов туристического агентства, зарегистрированных в системе. Класс имеет следующие атрибуты: *login* (имя пользователя), *password* (пароль), *fio* (фио), *address* (адрес клиента), *phone* (телефон), *факс* (fax). Класс находится в отношении с классом *Order*.
4. *CategoryOfTours* – класс, описывающий обслуживаемые категории туров. Класс имеет следующие атрибуты: *id* (идентификатор), *category* (категория тура), *price* (цена), *lengthy* (длительность проводимой акции).
5. *Action*s – класс, описывающий проводимые акции. Класс имеет следующие атрибуты: *id* (идентификатор), *name* (название акции), dateOfStart (дата начала акции), dateOfEnd (дата окончания акции), *description* (пояснения к акции).
6. *Order* – класс, описывающий заказы клиентов, хранящиеся в базе данных. Класс имеет следующие атрибуты: *id* (идентификатор), *country* (страна, выбранная клиентом), *city* (город, куда прибудет клиент), *hotel* (отель), dateOfSent (дата отправления), dateOfArrival (дата прибытия), *category* (категория тура), *price* (цена), *carrier* (перевозчик), *status* (статус оплачен/не оплачен). Класс находится в отношении ассоциации с классами *Actions*, *User*, *Client*, *Tours*.