Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный технологический университет»

Кафедра программной инженерии

«Проектирование интернет систем»

Лабораторная работа №7-8

**«Объектно-ориентированное моделирование. UML-диаграммы поведения»**

Выполнил:

студент 4 курса 5 группы

ФИТ

Почиковская Юлия Сергеевна

Проверил:

Пахолко А.С.

Минск 2022

**Цель:** Изучение методологии объектно-ориентированного моделирования средствами UML. Получение дополнительных навыков проектирования моделей информационной системы с применением возможностей UML диаграмм.

1. **Теоретические вопросы подготовки к лабораторной работе:**

1) Укажите виды диаграмм поведения. Какая между ними связь?

Диаграммы поведения в UML условно можно разделить на пять типов в соответствии с основными способами моделирования динамики системы.

Диаграммы прецедентов – описывают организацию поведения системы.

Диаграммы последовательностей – акцентируют внимание на времени упорядоченности сообщений.

Диаграммы кооперации сфокусированы на структурной организации объектов, посылающих и получающих сообщения.

Диаграммы состояний – описывают изменение состояния системы в ответ на события.

Диаграммы деятельности – демонстрируют передачу управления от одной деятельности к другой.

2) Опишите назначение диаграммы деятельности.

Диаграмма деятельности — UML-диаграмма, на которой показаны действия, состояния которых описаны на диаграмме состояний. Диаграммы деятельности используются при моделировании бизнес-процессов, технологических процессов, последовательных и параллельных вычислений.

3) Опишите основные нотации, которые используются на диаграмме состояний.

* класс(Class) – Пример;
* состояние(State) – Пример;
* состояние(StateEx) – Пример;
* составное состояние (Composite state) – Пример;
* разделитель (Concurrent state) – Пример;
* история (History) – ;
* глубокая история (Deep history) – Пример;
* начальное состояние (Start state) – Пример;
* конечное состояние (Final state) – Пример;
* синхронизатор/разветвитель (Complex transition) – Пример Пример;
* переход (Transition) – Пример;
* сообщение (Event message) – Пример;
* точка изгиба связей (Point) – Пример;
* комментарий (Note) – Пример;
* коннектор комментария (Note connector) – Пример;

4) Укажите виды связей между объектами на диаграмме последовательностей.

* Синхронное сообщение — отправитель передаёт ход управления актёру-получателю, которому необходимо провести в прецеденте некоторое действие. Пока проводимое получателем действие не будет завершено (соответственно, не будет получено ответное сообщение), отправитель теряет возможность производить какие-либо действия. Графически изображается как сплошная линия со стрелкой в виде закрашенного треугольника, после которой идёт прямоугольник, отражающий деятельность объекта, в конце которого находится ответное сообщение.
* Ответное сообщение — данное сообщение является ответом на синхронное сообщение. Обычно, содержит какое-либо возвращаемое изначальному отправителю значение, также возвращающее ему управление (возможность действовать). Графически изображается пунктирной линией с открытой стрелкой.
* Асинхронное сообщение — отправитель передаёт ход управления получателю, которому необходимо провести в прецеденте некоторое действие. Основное отличие от синхронного сообщения состоит в том, что отправитель не теряет возможности совершать другие действия. Графически изображается сплошной линией с открытой стрелкой.
* Потерянное сообщение — сообщение без адресата (есть отправитель, нет получателя).
* Найденное сообщение — сообщение без отправителя.

Последние два вида стрелок (взаимодействий) используются крайне редко. В основном они используются для демонстрации взаимодействия имеющихся объектов в данном прецеденте с внешними системами.

5) Какая диаграмма позволяет моделировать параллельные вычисления?

Диаграммы деятельности используются при моделировании бизнес-процессов, технологических процессов, последовательных и параллельных вычислений.

1. **Описание программно-аппаратных средств, используемые при выполнении работы:**

Построение моделей выполнялось в программной среде LucidChart.

LucidChart — это веб-приложение для построения диаграмм, которое позволяет пользователям визуально совместно рисовать, редактировать и обмениваться диаграммами, а также улучшать процессы, системы и организационные структуры. Он производится компанией Lucid Software Inc.

1. **Описание практического задания:**

Unified Modeling Language (UML) — унифицированный язык моделирования. Расшифруем: modeling подразумевает создание модели, описывающей объект. Unified (универсальный, единый) — подходит для широкого класса проектируемых программных систем, различных областей приложений, типов организаций, уровней компетентности, размеров проектов. UML описывает объект в едином заданном синтаксисе, поэтому где бы вы не нарисовали диаграмму, ее правила будут понятны для всех, кто знаком с этим графическим языком — даже в другой стране.

Одна из задач UML — служить средством коммуникации внутри команды и при общении с заказчиком. Давайте рассмотрим возможные варианты использования диаграмм:

* Проектирование. UML-диаграммы помогут при моделировании архитектуры больших проектов, в которой можно собрать как крупные, так и более мелкие детали и нарисовать каркас (схему) приложения. По нему впоследствии будет строиться код.
* Реверс-инжиниринг — создание UML-модели из существующего кода приложения, обратное построение. Может применяться, например, на проектах поддержки, где есть написанный код, но документация неполная или отсутствует.
* Из моделей можно извлекать текстовую информацию и генерировать относительно удобочитаемые тексты — документировать. Текст и графика будут дополнять друг друга.
  1. **Диаграмма деятельности**

Диаграммы деятельности – представляют собой графическое представление рабочих процессов поэтапных действий и действий с поддержкой выбора, итерации и параллелизма.

На данной диаграмме деятельности можем наблюдать процесс работы с получением баллов за сдачу отходов, он состоит из 5 шагов, следовательно, первым из них будет заполнение формы с количеством сданных отходов, далее отправка этих данных администратору для подтверждения. Администратор выполняет проверку подлинности предоставленного чека о сдаче отходов, если чек достоверен администратор подтверждает начисление баллов и эти баллы начисляются пользователь, в противном случаем пользователю будет отказано в начислении баллов.

В диаграмме деятельности применялись такие элементы как:

* принятие решения;
* активное состояние;
* начальное состояния;
* конечное состояние;
* объект в состоянии;
* переход.

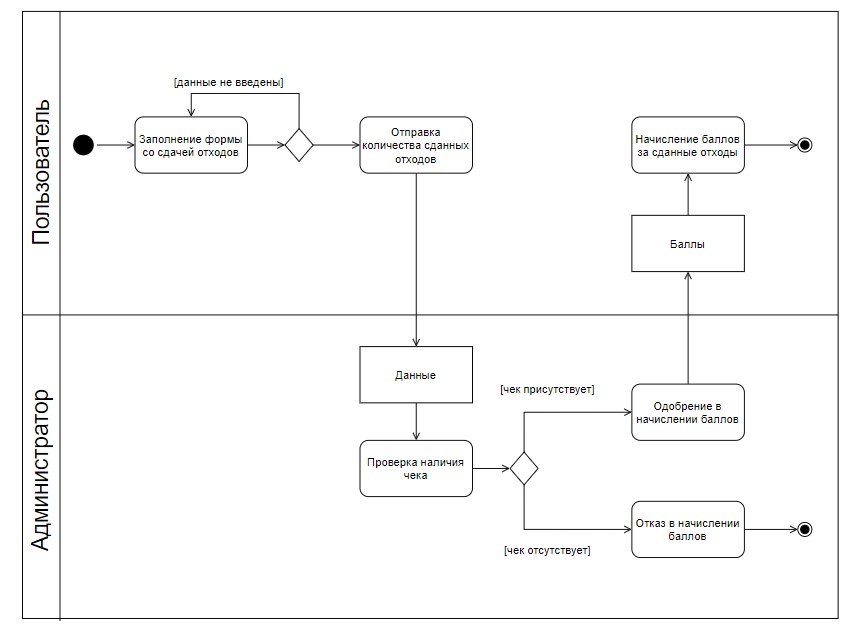


Рисунок 3.1 – UML Диаграмма деятельности

* 1. **Диаграмма состояний**

Диаграммы состояний служат для моделирования динамических аспектов системы. Данная диаграмма полезна при моделировании жизненного цикла объекта. От других диаграмм диаграмма состояний отличается тем, что описывает процесс изменения состояний только одного экземпляра определенного класса - одного объекта, причем объекта реактивного, то есть объекта, поведение которого характеризуется его реакцией на внешние события.

Для диаграммы состояний я выбрала непосредственно сам процесс начисления баллов. Первое состояние передача данных на проверку. После этого процесс ожидания начисления баллов, во время начисления может быть либо подтверждение начисления баллов либо отклонение.



Рисунок 3.1 – UML Диаграмма состояний

В диаграмме состояний применялись такие элементы как:

* класс;
* состояние;
* начальное состояние;
* конечное состояние;
* переход.