**ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ПРОФЕССИОНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ СРЕДНЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ МАРИЙ ЭЛ**

"**ЙОШКАР**-**ОЛИНСКИЙ** **ТЕХНОЛОГИЧЕСКИЙ** **КОЛЛЕДЖ**"

**ОТЧЁТ**

**О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ**

**«Диаграмма последовательности по теме Космос»**

Студента 3 курса, А-31 группы

Специальность 09.02.07 – «Информационные системы и программирования»

Квалификация «Администратор БД»

Руководитель: Пинешкин Ю.С.

Разработал:Тимофеев М.А.

г. Йошкар-Ола, 2022

**Диаграмма последовательности (англ. sequence diagram)** — UML-диаграмма, на которой для некоторого набора объектов на единой временной оси показан жизненный цикл объекта (создание-деятельность-уничтожение некой сущности) и взаимодействие актеров (действующих лиц) информационной системы в рамках прецедента.

Диаграммы последовательностей используются для уточнения диаграмм прецедентов, более детального описания логики сценариев использования. Это отличное средство документирования проекта с точки зрения сценариев использования!

Диаграммы последовательностей обычно содержат объекты, которые взаимодействуют в рамках сценария, сообщения, которыми они обмениваются, и возвращаемые результаты, связанные с сообщениями. Впрочем, часто возвращаемые результаты обозначают лишь в том случае, если это не очевидно из контекста.

Объекты обозначаются прямоугольниками с подчеркнутыми именами (чтобы отличить их от классов).

**Сообщения (вызовы методов)** - линиями со стрелками.

**Возвращаемые результаты** - пунктирными линиями со стрелками.

Прямоугольники на вертикальных линиях под каждым из объектов показывают “время жизни” (фокус) объектов. Впрочем, довольно часто их не изображают на диаграмме, все это зависит от индивидуального стиля проектирования.

Основные элементы диаграммы последовательности

Основными элементами диаграммы последовательности являются:

| **Элемент/Нотация** | **Предназначение** |
| --- | --- |
| Пример | Участник (Actor) |
| Пример | Объект (Object) |
| Пример | Активный объект (Active object) |
| Пример | Терминатор (Terminator) |
| Пример | Вызов процедуры (Procedure call) |
| Пример | Сообщение (Flat message) |
| Пример | Асинхронное сообщение (Async message) |
| Пример | Сообщение с результатом (Return message) |
| Пример | Временной интервал (In scope) |
| Пример | Временное ограничение (Time constraint) |
| Пример | Точка изгиба связей (Point) |
| Пример | Комментарий (Note) |
| Пример | Коннектор комментария (Note connector) |

Примеры диаграмм

*Виды стрелок*

Как было сказано выше, взаимодействие между акторами отображается при помощи специальных стрелок, передающих управление от отправителя (от кого идёт стрелка) к получателю (тот, к кому направлена стрелка). Стрелки демонстрируют ход сценария и те события, которые происходят во время анализируемого прецедента. Всего существует 5 видов стрелок:

* Синхронное сообщение — актор-отправитель передаёт ход управления актору-получателю, которому необходимо провести в прецеденте некоторое действие. Пока проводимое актором-получателем действие не будет завершено (соответственно, не будет получено ответное сообщение), актор-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 актору-отправителю значение, также возвращающее ему управление (возможность действовать). Графически изображается пунктирной линией с открытой стрелкой.
* Асинхронное сообщение — актор-отправитель передаёт ход управления актору-получателю, которому необходимо провести в прецеденте некоторое действие. Основное отличие от синхронного сообщения состоит в том, что актор-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.

Пример диаграммы последовательности

