

Специальность: 09.02.07 Информационные системы и программирование

Дисциплина/МДК: Разработка кода

**ОТЧЕТ**  
**По лабораторной работе № 2**  
**Тема: Построение диаграмм**  
**использования вариантов с помощью**  
**специализированных языков**

Выполнила студентка гр.И-9-23

\_\_\_\_\_

Семко М.В.

Оценка

\_\_\_\_\_

\_\_\_\_\_

(оценка прописью)

Проверил преподаватель

\_\_\_\_\_

Лихторенко О.С

г.Москва  
2024 г.

**Кратка теория:**

**Диаграмма прецедентов** - диаграмма, отражающая отношения между акторами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

**Тема: Диаграмма прецедентов**

**Цель:** Умение написания Диаграммы прецедентов

ТО и ПО: процессор: Intel Pentium G630, идеокарта: Intel 2nd Generation Core Processor Family Integrated Graphics Controller, ОС: Windows 10, ОЗУ: 2гб, Браузер: Chrome

**Порядок выполнения работы:**

1. Изучение теоретического материала.
2. Обзор на синтаксис и особенностей.
3. Обзор на онлайн - инструменты.
4. Практическая часть.

**Листинги:**

```
@startuml
actor Зам.Директора
actor Обучающийся
actor Преподаватель
actor Кл.Руководитель
(Зам.Директора) -up-> (Составить расписание)
(Составить расписание занятий) <.down. (Составить расписание) : include
(Составить расписание мероприятий) <.down. (Составить расписание) : include
(Составить расписание каникул) <.left. (Составить расписание) : include

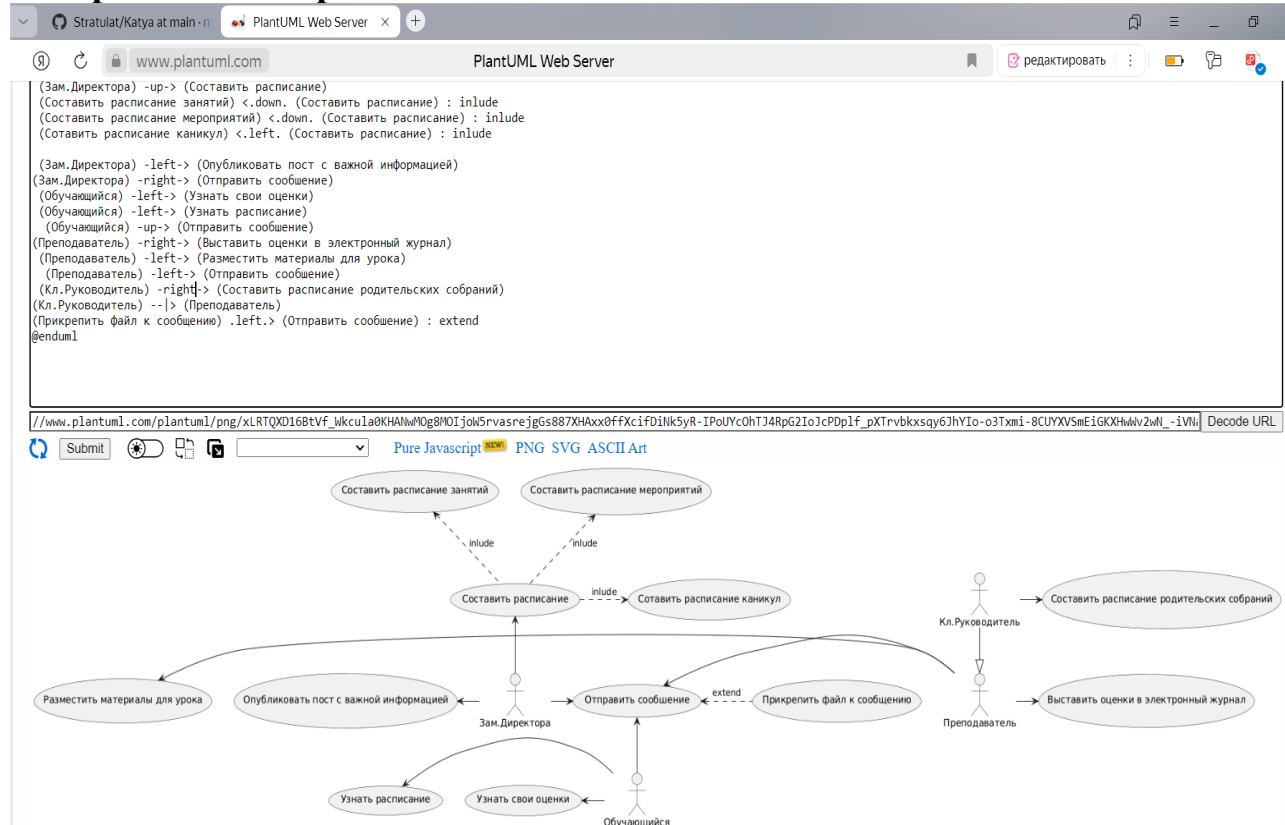
(Зам.Директора) -left-> (Опубликовать пост с важной информацией)
(Зам.Директора) -right-> (Отправить сообщение)
(Обучающийся) -left-> (Узнать свои оценки)
(Обучающийся) -left-> (Узнать расписание)
(Обучающийся) -up-> (Отправить сообщение)
(Преподаватель) -right-> (Выставить оценки в электронный журнал)
(Преподаватель) -left-> (Разместить материалы для урока)
(Преподаватель) -left-> (Отправить сообщение)
(Кл.Руководитель) -right-> (Составить расписание родительских собраний)
(Кл.Руководитель) --|> (Преподаватель)
(Прикрепить файл к сообщению) .left.> (Отправить сообщение) : extend
@enduml
```

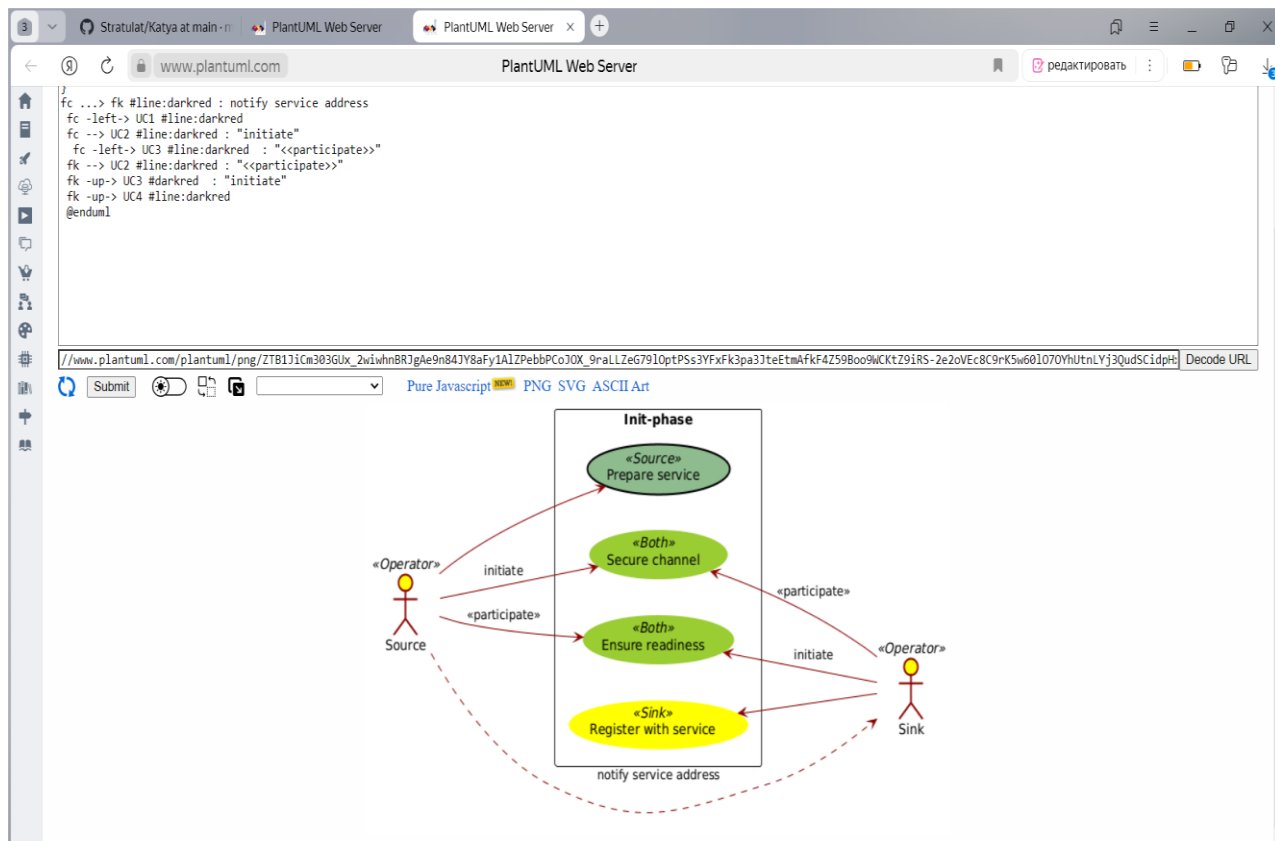
```

left to right direction
actor Source <<Operator>> as fc
#yellow;line:darkred;line.bold actor Sink <<Operator>> as
fk #yellow;line:darkred;line.bold rectangle Init-phase {
    usecase "Prepare service" <<Source>> as UC1 #darkseagreen;line:black;line.bold
    usecase "Secure channel" <<Both>> as UC2
    #yellowgreen;line:yellowgreen;line.bold usecase "Ensure readiness" <<Both>> as
    UC3 #yellowgreen;line:yellowgreen;line.bold usecase "Register with service"
    <<Sink>> as UC4 #yellow;line:yellow;line.bold
}
fc ...> fk #line:darkred : notify service
address fc -left-> UC1 #line:darkred
fc --> UC2 #line:darkred : "initiate"
fc -left-> UC3 #line:darkred :
" <<participate>>" fk --> UC2 #line:darkred
: " <<participate>>"
fk -up-> UC3 #darkred :
"initiate" fk -up-> UC4
#line:darkred @enduml

```

## Изображение диаграммы:





**Вывод: В ходе работы я приобрел опыт умения написания диаграммы прецендентов**