

**Министерство науки высшего образования Российской Федерации**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО**  
**ОБРАЗОВАНИЯ**  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**  
**(Университет ИТМО)**

**Факультет** информационных технологий и программирования

**Образовательная программа Программирование и интернет-технологии**

**Направление подготовки (специальность) 09.03.02 Информационные системы и технологии**

## **О Т Ч Е Т**

об учебной практике

Тема задания: Тема К: Создание консольного приложения на OSGi

Обучающийся Даутов Мурат Дамирович

Руководитель практики от профильной организации: компания SoftWerke

Руководитель практики от университета: Зубок Д.А., доцент ФИТИП

Практика пройдена с оценкой \_\_\_\_\_

Подписи членов комиссии:

\_\_\_\_\_ Зубок Д.А.  
(подпись)

\_\_\_\_\_ Маятин А.В.  
(подпись)

\_\_\_\_\_ Повышев В.В.  
(подпись)

Дата \_\_\_\_\_

Санкт-Петербург  
2020

## 1 этап

Для установки Felix Web Console я установил Apache Felix EventAdmin, Apache Felix Http Jetty, Apache Felix Servlet API, Apache Felix Web Management Console (All In One).

### Apache Felix Web Console Bundles



Main

OSGi

Status

Web Console

Log out

Bundle information: 16 bundles in total, 14 bundles active, 2 bundles installed

✖

Apply Filter

Filter All

Reload

Install/Update...

Refresh Packages

<div>Id</div>	<div>Name</div>	<div>Version</div>	<div>Category</div>	<div>Status</div>	<div>Actions</div>
0	<div>▶ System Bundle <i>(org.apache.felix.framework)</i></div>	6.0.3		Active	<div><div></div><div></div><div></div></div>
1	<div>▶ jansi <i>(org.fusesource.jansi)</i></div>	1.17.1		Active	<div><div></div><div></div><div></div></div>
2	<div>▶ JLine Bundle <i>(org.jline)</i></div>	3.7.0		Active	<div><div></div><div></div><div></div></div>
3	<div>▶ Apache Felix Bundle Repository <i>(org.apache.felix.bundlerepository)</i></div>	2.0.10		Active	<div><div></div><div></div><div></div></div>
4	<div>▶ Apache Felix Gogo Command <i>(org.apache.felix.gogo.command)</i></div>	1.0.2		Active	<div><div></div><div></div><div></div></div>
5	<div>▶ Apache Felix Gogo JLine Shell <i>(org.apache.felix.gogo.jline)</i></div>	1.1.0		Active	<div><div></div><div></div><div></div></div>
6	<div>▶ Apache Felix Gogo Runtime <i>(org.apache.felix.gogo.runtime)</i></div>	1.1.0		Active	<div><div></div><div></div><div></div></div>
7	<div>▶ Apache Felix EventAdmin <i>(org.apache.felix.eventadmin)</i></div>	1.5.0		Active	<div><div></div><div></div><div></div></div>
8	<div>▶ Apache Felix Configuration Admin Service <i>(org.apache.felix.configadmin)</i></div>	1.9.16	osgi	Active	<div><div></div><div></div><div></div></div>
9	<div>▶ Apache Felix Servlet API <i>(org.apache.felix.http.servlet-api)</i></div>	1.1.2		Active	<div><div></div><div></div><div></div></div>
10	<div>▶ Apache Felix Http Jetty <i>(org.apache.felix.http.jetty)</i></div>	4.0.18		Active	<div><div></div><div></div><div></div></div>
13	<div>▶ Apache Felix Web Management Console (All In One) <i>(org.apache.felix.webconsole)</i></div>	4.5.2.all		Active	<div><div></div><div></div><div></div></div>

✖

Apply Filter

Filter All

Reload

Install/Update...

Refresh Packages

Bundle information: 16 bundles in total, 14 bundles active, 2 bundles installed

## 2 этап

<https://github.com/Murakle/SummerPracticePart2> (В папке result хранятся готовые jar файлы)

HelloWorldPrinter зависит только от osgi фреймворка. В HelloWorldPrinter есть сервис PrintHelloWorld в котором есть метод printHelloWorld().

SimpleBundle зависит от osgi фреймворка и HelloWorldPrinter. При активации бандла вызывается метод printHelloWorld().

Регистрация сервиса:

```
serviceRegistration = (ServiceRegistration<IPrintHelloWorld>)
    context.registerService(IPrintHelloWorld.class.getName(),
        new PrintHelloWorldImpl(),
        null);
```

Демонстрация работы

```
[g! install /Users/user/dev/ITMO/SummerPractice/Part3/SimpleBundle/target/SimpleB
undle-1.0.0.jar
Bundle ID: 11
[g! start 11
Hello OSGi World!
g! 12:07:03]
12:07:11
```

## 3 этап

<https://github.com/Murakle/SummerPracticePart3>

Все также, как и в предыдущем этапе, но добавились зависимости от org.osgi.service.component.annotations и org.apache.felix.scr

В сервисе PrintHelloWorld добавились аннотации @Component, @Activate и @Deactivate

Пример аннотации:

```
@Component(service = IPrintHelloWorld.class)
```

## 4 этап

<https://github.com/Murakle/SummerPracticePart4>

Для создания собственной команды так же необходим бандл org.apache.felix.gogo.runtime а так же зависимости org.osgi.service.component.annotations и org.apache.felix.scr.

Для создание команды practice:hello в активаторе регистрируем команду как обычный сервис. А в самой команде с помощью аннотаций помечаем о том что это команда:

```
@Component(  
    property = {  
        CommandProcessor.COMMAND_SCOPE + ":String=practice",  
        CommandProcessor.COMMAND_FUNCTION + ":String=hello"  
    },  
    service = HelloCommand.class  
)  
  
public class HelloCommand {...}
```

В HelloCommand должна быть функция совпадающая с параметром COMMAND\_FUNCTION. В данном случае hello(String param);

```
[g! install /Users/user/dev/ITMO/SummerPractice/Part4/result/practice-1.0.0.jar ]  
Bundle ID: 12  
[g! start 12 12:20:07]  
[g! practice:hello abacaba 12:20:10]  
Hello abacaba  
g! █ 12:20:21
```

## 5 этап

<https://github.com/Murakle/SummerPracticePart5>

Задание реализовано как один проект с несколькими модулями, каждый из которых реализует отдельный бандл.

### Модули:

Модуль Request – модуль с одним интерфейсом IRequest в котором лишь одна функция getTitles().

Модуль Application – основной модуль, который зависит от модуля Request и в нем реализованы три команды news:add, news:remove, news:stats. Так же в нем есть сервис (SourceService), в котором хранятся зарегистрированные источники.

Модули LentaRequest и Meduza Request – модули зависящие от модуля Request, в каждом из которых регистрируется один сервис, который является реализацией интерфейса IRequest.

Для Http запроса используются встроенные в java инструменты, а для чтения возвращаемого json файла использовался бандл org.json. При регистрации сервисов, к ним прикреплялся параметр “source=<название\_источника>”

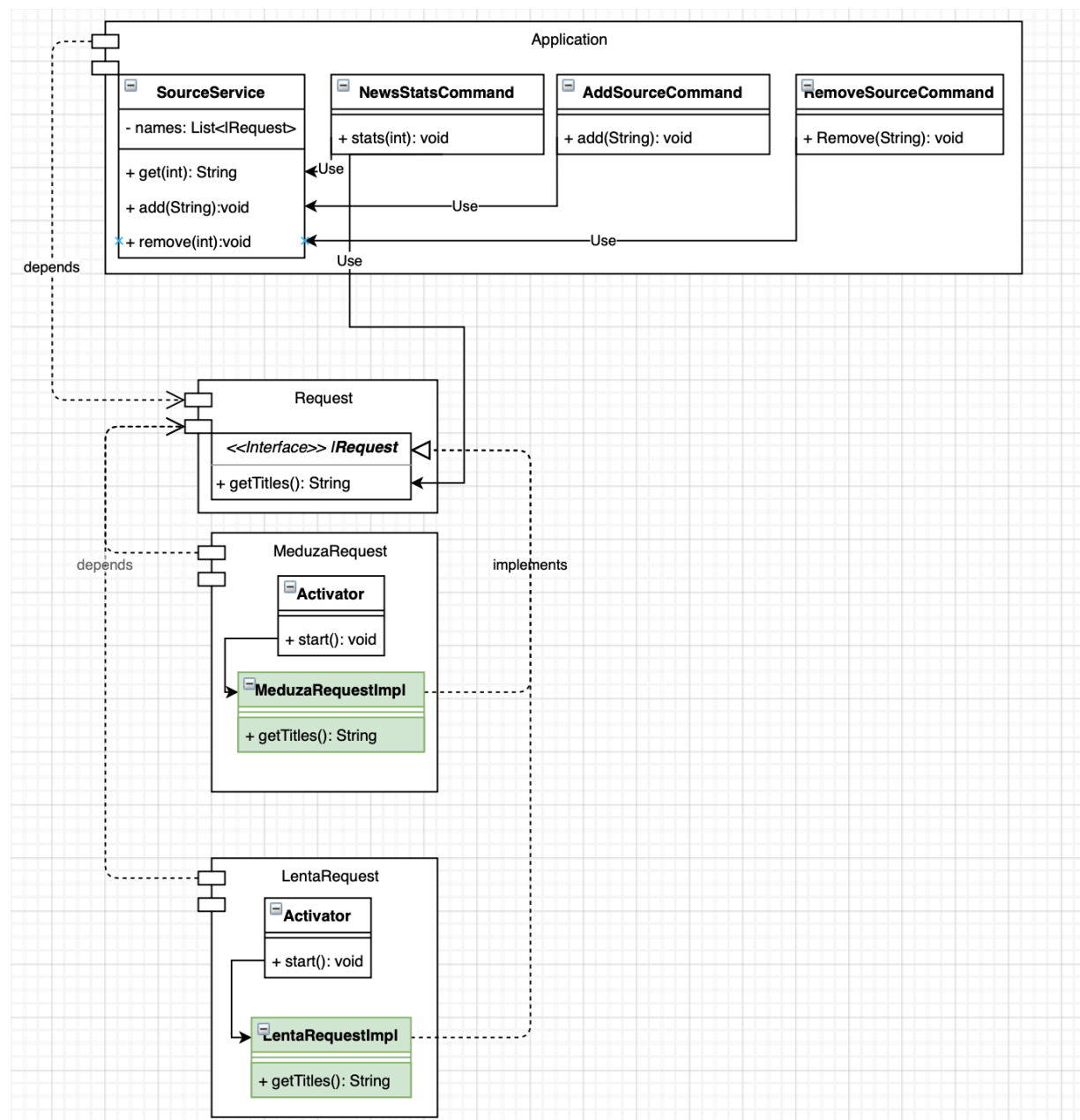
### Команды:

news:add path – устанавливает бандл, который находится в переданной строке и запускает его, тем самым регистрируя сервисы имплементирующие IRequest. А затем добавляет эти сервисы в SourceService.

news:remove index – берет источник(сервис), который находится под переданным индексом удаляет его из SourceService и удаляет бандл содержащий его.

news:stats index – берет источник находящийся под переданным индексом и запрашивает у него все заголовки. Затем подсчитывает количество слов и выводит 10 самых популярных.

### Диаграмма:



### Пример использования:

```
[g! news:add file:/Users/user/dev/ITMO/SummerPractice/Part5/MyApp/LentaRequest/target/LentaRequest-1.0.0.jar
Start LentaRequest activator
[g! news:stats
Choose sources from given:
0. All
1. Meduza
2. Lenta
2
России 8
США 5
Власти 3
Мишустин 3
Роскосмоса 3
Украина 3
ФСБ 3
борьбе 3
видео 3
главы 3
g! █
```

15:22:52]

15:23:07

### Вывод:

Плюсы OSGI:

1. Динамические обновления – можно устанавливать, обновлять, удалять без остановки приложения
2. Упрощение разработки приложения – благодаря модульности разрабатывать большие приложения намного легче
3. Управление версиями – разные версии одной библиотеки, могут быть загружены разные версии одной и той же библиотеки.

Минусы:

1. Нет интеграции со всеми библиотеками

Данную технологию удобно применять, когда нужно, что бы дополнительные модули добавлялись во время исполнения программы. И не нужно использовать, когда требуется использовать сторонние библиотеки, не имеющие osgi-bundle'ов. Так же