

ООО "Софтверке"

Отчет по практике

Студент группы Р3317

Пышный Максим Александрович

Санкт-Петербург

2017

Этап 2

Описание всех классов и пакетов

Структура пакета **org.kelinago.osgi.provider**:

- **able** – пакет, содержащий интерфейс, реализуемый сервисом
 - **Provider.java** - интерфейс
- **impl** – пакет, содержащий сервис, реализующий интерфейс
 - **ProviderImpl.java** - Сервис
- **ServiceActivator.java** – активатор сервиса

Структура пакета **org.kelinago.osgi.consumer**:

- **Consumer.java** – потребитель сервиса
- **Activator.java** – активатор бандла

Описание бандлов

Task2-Provider

- Бандл, предоставляющий сервис
- Не имеет сторонних зависимостей

Task2-Consumer

- Бандл, потребляющий сервис
- Требуется наличие бандла Task2-Provider

Ссылка на git репозиторий

- [GitHub](#)

Ссылки на собранные бандлы

- [Task2-Provider-1.0.jar](#)
- [Task2-Consumer-1.0.jar](#)

Этап 3

Описание всех классов и пакетов

Структура пакета **com.kelinago.samples.osgi**:

- **able** – пакет, содержащий интерфейс, реализуемый сервисом
 - **Greeter.java** – интерфейс
- **SimpleGreeter.java** - сервис
- **Consumer.java** – потребитель сервиса

Описание бандлов

Task3-Service

- Бандл, предоставляющий сервис
- Не имеет сторонних зависимостей
- Для корректного запуска необходимо наличие бандла org.apache.felix.scr

Task3-Consumer

- Бандл, потребляющий сервис
- Требуется наличие бандла Task3-Service
- Для корректного запуска необходимо наличие бандла org.apache.felix.scr

Ссылка на архив с исходным кодом или ссылка на открытый git репозиторий

- [GitHub](#)

Ссылки на собранные бандлы

- [Task3-Service-1.0.jar](#)
- [Task3-Consumer-1.0.jar](#)

Этап 4

Описание всех классов и пакетов

Структура пакета **practice.task4**:

- **able** – пакет, содержащий интерфейс, реализуемый сервисом
 - **Greeter.java** – интерфейс
- **NameGreeter.java** - сервис, используемый в качестве команды GoGo Shell

Описание бандлов

Task4

- Бандл, регистрирующийся в качестве команды
- Не имеет сторонних зависимостей
- Для корректного запуска необходимо наличие бандла org.apache.felix.scr

Ссылка на архив с исходным кодом или ссылка на открытый git репозиторий

- [GitHub](#)

Ссылки на собранные бандлы

- [Task4-1.0.jar](#)

Этап 5

Описание всех классов и пакетов

Структура пакета **practice.task5**:

- **api** – пакет, содержащий интерфейс, реализуемый сервисами
 - **NewsTitles.java** – интерфейс
- **impl** – пакет, содержащий реализации интерфейса
 - **LentaNews.java** – сервис получения заголовков новостей из Lenta.ru
 - **AIFNews.java** - сервис получения заголовков новостей из RSS АиФ
- **Command.java** - сервис, используемый в качестве команды GoGo Shell
- **CommandActivator.java** – активатор бандла команды

Описание бандлов

Command

- Бандл, регистрирующийся в качестве команды
- Требуется наличие бандла stats-api
- Для корректного запуска необходимо наличие бандла org.apache.felix.scr

lenta-news

- Бандл, предоставляющий сервис получения заголовков новостей из источника Lenta.ru
- Имеет встроенную стороннюю зависимость json-simple
- Требуется наличие бандла stats-api
- Для корректного запуска необходимо наличие бандла org.apache.felix.scr

aif-news

- Бандл, предоставляющий сервис получения заголовков новостей из источника RSS АиФ
- Не имеет сторонних зависимостей
- Требуется наличие бандла stats-api
- Для корректного запуска необходимо наличие бандла org.apache.felix.scr

stats-api

- Бандл, предоставляющий интерфейс
- Не имеет сторонних зависимостей

Ссылка на архив с исходным кодом или ссылка на открытый git репозиторий

- [GitHub](#)

Ссылки на собранные бандлы

- [Command-1.0.jar](#)
- [lenta-news-1.0.jar](#)
- [aif-news-1.0.jar](#)
- [stats-api-1.0.jar](#)

Выводы

Главным достоинством OSGi является модульность, предоставляющая возможности динамической шины для Java-приложений.

Главным недостатком я бы назвал повышенную сложность обеспечения приемлемой изоляции модулей.

На OSGi целесообразно реализовывать:

- Системы «Умного дома»
- Web-приложения
- и т.д.