**Министерство образования и науки Российской Федерации**

федеральное государственное автономное образовательное учреждение высшего образования

**“НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**УНИВЕРСИТЕТ ИТМО**”

**Факультет программной инженерии и компьютерной техники**

**ЛАБОРАТОРНАЯ РАБОТА № 7**

**ПО ДИСЦИПЛИНЕ «ТЕХНОЛОГИИ ВЕБ-СЕРВИСОВ»**

Студент: Носков Фёдор Андреевич

Группа: P41142

Преподаватель: Дергачев Андрей Михайлович

Санкт-Петербург

2020

1.Техническое задание

Требуется разработать приложение, осуществляющее регистрацию сервиса в реестре jUDDI, а также поиск сервиса в реестре и обращение к нему. Рекомендуется реализовать консольное приложение, которое обрабатывает 2 команды. Итог работы первой команды – регистрация сервиса в реестре; вторая команда должна осуществлять поиск сервиса, а также обращение к нему.

2.Описание работы

Было разработано консольное приложении удовлетворяющее условием задачи. Сервер jUDDI разворачивался на виртуальной машине с операционной системой Ubuntu.

Консольное приложение расположно на GitHub по адресу: https://github.com/xim123456/WebLabs

3.Ответы на вопросы

1. В каком случае вы предпочтете сервис-ориентированную архитектуру вместо монолитной? Приведите ваши рассуждения по данному вопросу.

Сервис-ориентированная архитектура имеет следующие преимущества:

1. Легкость в сопровождении так как размер каждого сервиса меньше монолита. Возможность использовать множество разных инструментов.
2. Более высокая надежность, так как службы легче отлаживать и тестировать, чем огромные куски кода, как в монолитах.
3. Возможность параллельной разработки, так как программ множество, а не одна.

Следовательно, сервис-ориентированную архитектуру нужно использовать в больших проектах, где требуется множество разного функционала и подразумевается обслуживание большого количество людей.

1. В чем заключается смысл использования UDDI-реестров в промышленных SOA-системах? Можно ли обойтись без реестров?

UDDI-реестры необходимы чтобы хранить информацию о сотнях промышленных сервисах и предоставлять техническую информацию о них. Так же эти реестры предоставляют удобный механизм поиска по этим сервисам. Так как современные системы очень быстро развиваются, новые сервисы появляются, а в старых меняются интерфейсы просто необходимо иметь реестр для хранения информации о них.

Возможно можно обойтись без UDDI-реестров используя внутреннюю документацию компании или хранит информацию о сервисах в подобных системах. Но искать сервисы в подобных подходах наверняка будет не так удобно.

1. Для чего используются такие компоненты SOA, как ESB? Можно ли обойтись без них?

Enterprise Service Bus — связующее программное обеспечение, обеспечивающее централизованный и унифицированный событийно-ориентированный обмен сообщениями между различными информационными системами на принципах сервис-ориентированной архитектуры.

Из этого следует что ESB инструмент который отвечает за маршрутизацию сообщений, оркестровку сервисов и надёжность доставки самих сообщений.

Возможно можно обойтись без ESB если заранее расписать форматы и маршруты доставки всех сообщений в системе или как-то унифицировать общение всех сервисов. Но на практике это редко бывает возможно из-за большой скорости рост системы.

1. Предположим Вам нужно спроектировать public API для вашей системы. Что вы выберете, REST или SOAP? Какие будут основные критерии для выбора? Приведите ваши рассуждения по данному вопросу.

Всё зависит от требований предоставленных к этому API. Если важны такие критерии как скорость доставки и объём трафика и простой интерфейс, то следует использовать REST, так как его подход к структуре сообщений более гибкий и экономный. Если же API будет иметь сотни сервисов и требовать интеграции с другими сервисами, то SOAP с его WSDL будет гораздо проще интегрировать и использовать.