**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 № 1**

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

Студент: Носков Антон Андреевич

Группа: P41142

Преподаватель: Дергачев Андрей Михайлович

Санкт-Петербург

2020

1.Техническое задание

В данной работе требуется создать таблицу в БД, содержащую не менее 5 полей, а также реализовать возможность поиска по любым комбинациям полей с помощью SOAP-сервиса. Данные для поиска должны передаваться в метод сервиса в качестве аргументов. Веб-сервис необходимо реализовать в виде standalone-приложения и J2EE-приложения. При реализации в виде J2EE-приложения следует на стороне сервера приложений настроить источник данных, и осуществлять его инъекцию в коде сервиса. Для демонстрации работы разработанных сервисов следует также разработать и клиентское консольное приложение.

2.Описание работы

Сервер написан на ejb, с использованием jdbc для подключения к базе данных. В качестве базы данных используется MySQL. Таблица использованная в лабораторной работе называется People и состоит из 5 полей: Id: int, Name: varchar(20), SecondName: varchar(20), Age: int, Sex: varchar(5).

Лабораторная работа и консольный клиент расположены на GitHub по адресу: <https://github.com/anton-mix/WebLabs>

3.Вопросы на защиту

1. Как я увидел, в реализации клиента Вы пишете request руками в сокет. Это работает, но, не является корректной реализацией SOAP-клиента. Как можно сгенерировать клиента с помощью wsimport? Используйте это в ваше работе.

Да, возможно сгенерировать все необходимые артефакты с помощью утилиты wsimport. Эта утилита предназначена для создания JAX-WS артефактов. При её вызове будут созданы:

* 1. Интерфейс описывающий необходимые методы
  2. Сервис эти методы реализующий
  3. Классы Exception, если в wsdl был тег fault
  4. И классы необходимые для общенияс серивисом

Вызвать wsimport можно двумя способами:

1. Из консоли, передав в аргументы утилите ссылку на wsdl
2. Или использовать IDE и работать с этой утилитой через gui

В 3 лабораторной работе использована утилита wsimport.

1. Что такое JAXB, как аннотации JAXB можно применить в контексте SOAP-сервисов?

JAXB(Java Architecture for XML Binding) – инструмент который позволяет сериализовать java класс в XML и наоборот. JAXB является частью jdk, по этому его не надо устанавливать дополнительно. Работает при помощи специальных аннотаций, которые показывают, как и что сериализовать.

В контексте SOAP это очень удобный инструмент. Так как SOAP для общения использует формат XML необходим инструмент, с помощью которого можно создавать XML сообщения, содержащие java объекты и конвертировать XML ответ в java объекты.

1. Предположим, при реализации SOAP веб-сервиса Вам необходимо передавать двоичные данные (аватары пользователей, архивы...). Какие стратегии Вы можете предложить для работы с binary attachments? Реализуйте одну из них.

Возможны две стратегии передачи двоичных данных:

1. Передача через аргументы SOAP сообщения обычным массивом byte. Но при такой стратегии невозможно будет передавать файлы большого размера.
2. Передача с помощью MTOM. MTOM позволяет SOAP отправлять бинарные файлы. В тело сообщения на место файла ложится специальный идентификатор файла, а сами бинарные данные идут после soap сообщения.

Я воспользовался второй стратегией.

1. Обычно, взаимодействие со внешним сервисом является дорогостоящей операцией, и может привести к значительным задержкам. В таких случаях имеет смысл взаимодействовать с веб-сервисом асинхронно. Модифицируйте клиентское приложение таким образом, чтобы взаимодействие с веб-сервисом было асинхронным.

Для того чтобы взаимодействовать с сервером асинхронно необходимо вынести взаимодействие в отдельный поток.

В модифицированной работе присутствуют два потока. Первый взаимодействует с сервером. Для того чтобы это было видно в каждую операцию сервера был добавлен sleep(5000). Второй поток запускает выполнение первого и начинает бесконечно проверять состояние переменных, в которые первый поток положит ответ. В это же время он выполняет полезную работу, в моём случае просто отправляет в консоль сообщение каждую секунду.