Министерство науки и высшего образования Российской Федерации

**Федеральное государственное бюджетное образовательное учреждение науки и высшего образования**

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

**(ВлГУ)**

**Кафедра информационных систем и программной инженерии**

Лабораторная работа №7

по дисциплине

" Интеграция кроссплатформенных программных систем"

Тема: " Разработка веб-сервиса средствами JAX-WS"

Выполнил:

ст. гр. ИСТм-121

В.Г. Хлызова

Принял:

Спирин И.В.

Владимир, 2022 г.

ЦЕЛЬ РАБОТЫ

Разработать сессионный компонент без состояния с представлением в виде веб-сервиса. Создать клиент для веб-сервиса.

ХОД РАБОТЫ

Средства выполнения:

1. JDK 17
2. NetBeans IDE 14
3. Glassfish Server 6.2.5

Разработанный веб-сервис и клиент: <https://github.com/arranay/Integration_of_cross_platform_software_systems/tree/main/lab7>

Создание веб-клиента

1. Устанавливаем NetBeans IDE 14 и создаем веб приложение

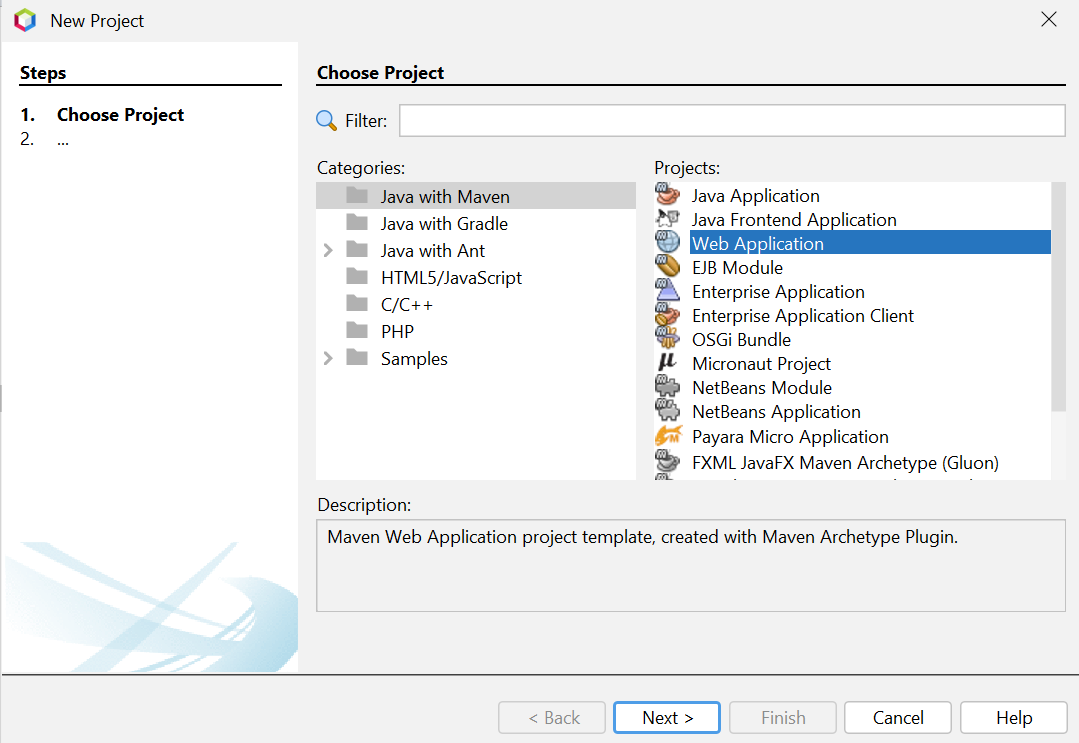


Рисунок 1 - Создание веб-приложения

1. Создаем классы для описания моделей данных, которые будут использоваться в нашем веб сервисе, используя аннотации из библиотеки jakarta.xml.bind.annotation (Рисунок 2)

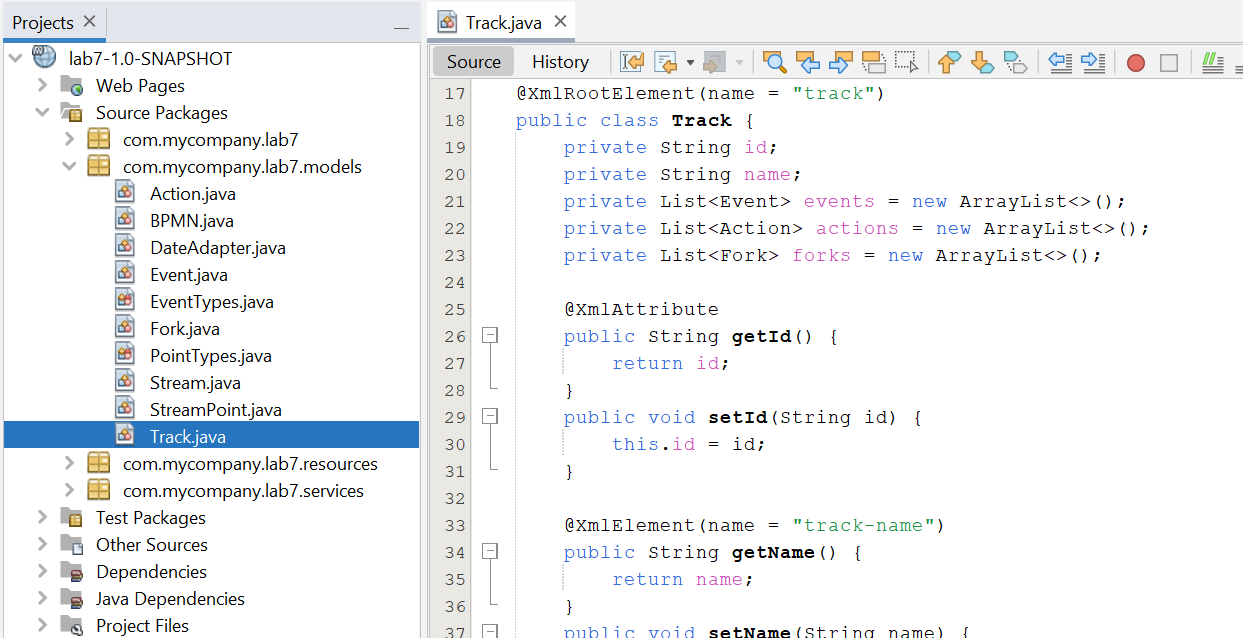


Рисунок 2 - Описание моделей.

1. Создаем описание методов веб-сервиса.

package com.mycompany.lab7.services;

import com.mycompany.lab7.models.Track;

import jakarta.jws.WebMethod;

import jakarta.jws.WebService;

import jakarta.jws.soap.SOAPBinding;

/\*\*

 \*

 \* @author valer

 \*/

@WebService

@SOAPBinding(style = SOAPBinding.Style.RPC)

public interface BpmnService {

    @WebMethod

    public boolean addTrack(Track track);

    @WebMethod

    public boolean deleteTrack(String id);

    @WebMethod

    public Track getTrack(String id);

    @WebMethod

    public Track[] getAllTracks();

}

1. Создаем класс с реализацией методов веб-сервиса.

package com.mycompany.lab7.services;

import com.mycompany.lab7.models.Track;

import jakarta.ejb.EJB;

import jakarta.jws.WebService;

import java.util.List;

/\*\*

\*

\* @author valer

\*/

@WebService(endpointInterface = "com.mycompany.lab7.services.BpmnService")

public class BpmnServiceImplementation implements BpmnService {

@EJB

private BpmnSessionBeanInterface bpmnSessionBean;

@Override

public boolean addTrack(Track track) {

Track result = bpmnSessionBean.addTrack(track);

return result != null;

}

@Override

public boolean deleteTrack(String id) {

int number = bpmnSessionBean.getTracks().size();

List<Track> list = bpmnSessionBean.deleteTrack(id);

return (number - list.size()) == 1;

}

@Override

public Track getTrack(String id) {

return this.bpmnSessionBean.findTrack(id);

}

@Override

public Track[] getAllTracks() {

List<Track> cityList = this.bpmnSessionBean.getTracks();

Track[] cities = new Track[cityList.size()];

Track[] array = cityList.toArray(cities);

return array;

}

}

1. Создаем сессионный компонент для реализации логики работы с объектом Track:

package com.mycompany.lab7.services;

import com.mycompany.lab7.models.Action;

import com.mycompany.lab7.models.Event;

import com.mycompany.lab7.models.EventTypes;

import com.mycompany.lab7.models.Fork;

import com.mycompany.lab7.models.Track;

import jakarta.ejb.Stateless;

import java.util.ArrayList;

import java.util.List;

/\*\*

 \*

 \* @author valer

 \*/

@Stateless

public class BpmnSessionBean implements BpmnSessionBeanInterface {

    List<Track> tracks;

    @Override

    public Track addTrack(Track track) {

        if (tracks == null)

            tracks = new ArrayList<>();

        this.tracks.add(track);

        return findTrack(track.getId());

    }

    @Override

    public List<Track> getTracks() {

        if (tracks == null) {

            tracks = new ArrayList<>();

            Track track = createFirstTrack();

            tracks.add(track);

        }

        return this.tracks;

    }

    @Override

    public Track findTrack(String id) {

        for(Track track : tracks) {

            if(track.getId().equals(id)) {

                return track;

            }

        }

        return null;

    }

    @Override

    public List<Track> deleteTrack(String id) {

        Track track = findTrack(id);

        if (track != null){

            tracks.remove(track);

        }

        return getTracks();

    }

    private Track createFirstTrack() {

        Track newTrack = new Track();

        newTrack.setId("123");

        newTrack.setName("name");

        List<Event> events = new ArrayList<>();

        Event event = new Event();

        event.setId("1");

        event.setName("name");

        event.setType(EventTypes.start);

        events.add(event);

        List<Action> actions = new ArrayList<>();

        Action action = new Action();

        action.setId("2");

        action.setOperation("start");

        action.setTime(12);

        actions.add(action);

        List<Fork> forks = new ArrayList<>();

        Fork fork = new Fork();

        fork.setCondition("condition");

        fork.setId("3");

        forks.add(fork);

        newTrack.setEvents(events);

        newTrack.setActions(actions);

        newTrack.setForks(forks);

        return newTrack;

    }

}

1. Устанавливаем Glassfish Server 6.2.5, собираем созданное приложение с помощью NetBeans IDE 14 и разворачиваем его (Рисунок 3).

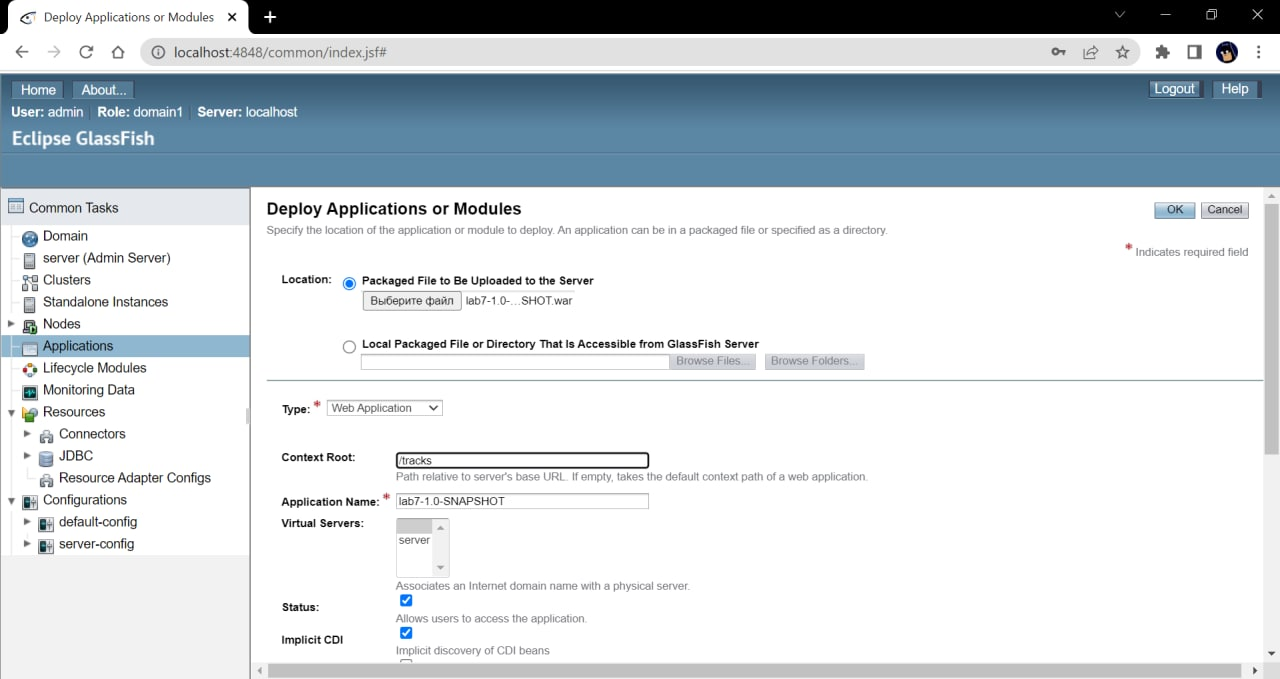


Рисунок 3 - Разворачиваем приложение.

Сгенерированный wsdl файл:

<definitions xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" xmlns:wsp="http://www.w3.org/ns/ws-policy" xmlns:wsp1\_2="http://schemas.xmlsoap.org/ws/2004/09/policy" xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://services.lab7.mycompany.com/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://schemas.xmlsoap.org/wsdl/" targetNamespace="http://services.lab7.mycompany.com/" name="BpmnServiceImplementationService">

<types>

<xsd:schema>

<xsd:import namespace="http://services.lab7.mycompany.com/" schemaLocation="http://laptop-mh9g0aui:8080/tracks/BpmnServiceImplementationService?xsd=1"/>

</xsd:schema>

</types>

<message name="addTrack">

<part name="arg0" type="tns:track"/>

</message>

<message name="addTrackResponse">

<part name="return" type="xsd:boolean"/>

</message>

<message name="deleteTrack">

<part name="arg0" type="xsd:string"/>

</message>

<message name="deleteTrackResponse">

<part name="return" type="xsd:boolean"/>

</message>

<message name="getTrack">

<part name="arg0" type="xsd:string"/>

</message>

<message name="getTrackResponse">

<part name="return" type="tns:track"/>

</message>

<message name="getAllTracks"/>

<message name="getAllTracksResponse">

<part name="return" type="tns:trackArray"/>

</message>

<portType name="BpmnService">

<operation name="addTrack">

<input wsam:Action="http://services.lab7.mycompany.com/BpmnService/addTrackRequest" message="tns:addTrack"/>

<output wsam:Action="http://services.lab7.mycompany.com/BpmnService/addTrackResponse" message="tns:addTrackResponse"/>

</operation>

<operation name="deleteTrack">

<input wsam:Action="http://services.lab7.mycompany.com/BpmnService/deleteTrackRequest" message="tns:deleteTrack"/>

<output wsam:Action="http://services.lab7.mycompany.com/BpmnService/deleteTrackResponse" message="tns:deleteTrackResponse"/>

</operation>

<operation name="getTrack">

<input wsam:Action="http://services.lab7.mycompany.com/BpmnService/getTrackRequest" message="tns:getTrack"/>

<output wsam:Action="http://services.lab7.mycompany.com/BpmnService/getTrackResponse" message="tns:getTrackResponse"/>

</operation>

<operation name="getAllTracks">

<input wsam:Action="http://services.lab7.mycompany.com/BpmnService/getAllTracksRequest" message="tns:getAllTracks"/>

<output wsam:Action="http://services.lab7.mycompany.com/BpmnService/getAllTracksResponse" message="tns:getAllTracksResponse"/>

</operation>

</portType>

<binding name="BpmnServiceImplementationPortBinding" type="tns:BpmnService">

<soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="rpc"/>

<operation name="addTrack">

<soap:operation soapAction=""/>

<input>

<soap:body use="literal" namespace="http://services.lab7.mycompany.com/"/>

</input>

<output>

<soap:body use="literal" namespace="http://services.lab7.mycompany.com/"/>

</output>

</operation>

<operation name="deleteTrack">

<soap:operation soapAction=""/>

<input>

<soap:body use="literal" namespace="http://services.lab7.mycompany.com/"/>

</input>

<output>

<soap:body use="literal" namespace="http://services.lab7.mycompany.com/"/>

</output>

</operation>

<operation name="getTrack">

<soap:operation soapAction=""/>

<input>

<soap:body use="literal" namespace="http://services.lab7.mycompany.com/"/>

</input>

<output>

<soap:body use="literal" namespace="http://services.lab7.mycompany.com/"/>

</output>

</operation>

<operation name="getAllTracks">

<soap:operation soapAction=""/>

<input>

<soap:body use="literal" namespace="http://services.lab7.mycompany.com/"/>

</input>

<output>

<soap:body use="literal" namespace="http://services.lab7.mycompany.com/"/>

</output>

</operation>

</binding>

<service name="BpmnServiceImplementationService">

<port name="BpmnServiceImplementationPort" binding="tns:BpmnServiceImplementationPortBinding">

<soap:address location="http://laptop-mh9g0aui:8080/tracks/BpmnServiceImplementationService"/>

</port>

</service>

</definitions>

Получение всех сущностей

<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">

    <Body>

        <getAllTracks xmlns="http://services.lab7.mycompany.com/"/>

    </Body>

</Envelope>

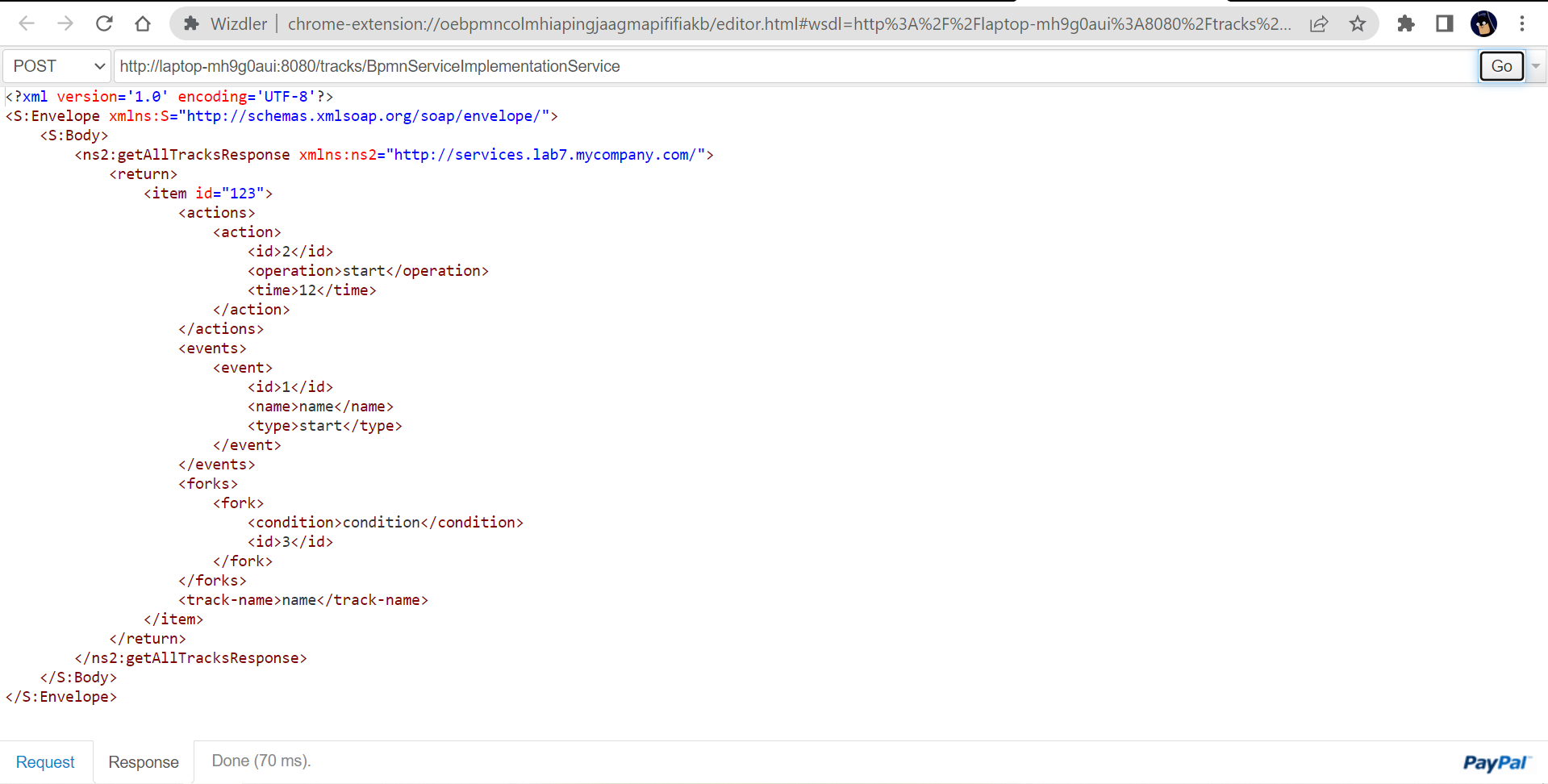


Рисунок 4 - Получение всех сущностей.

<?xml version="1.0" encoding="UTF-8"?>

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">

    <SOAP-ENV:Header/>

    <S:Body>

        <ns2:getTrack xmlns:ns2="http://services.lab7.mycompany.com/">

            <arg0>123</arg0>

        </ns2:getTrack>

    </S:Body>

</S:Envelope>

Получение сущности по идентификатору



Рисунок 5 - Получение сущности по идентификатору.

Добавление новой сущности по идентификатору

<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">

    <SOAP-ENV:Header/>

    <S:Body>

        <ns2:addTrack xmlns:ns2="http://services.lab7.mycompany.com/">

            <arg0 id="111">

                <actions>

                    <action>

                        <id>123</id>

                        <operation>start</operation>

                        <time>12</time>

                    </action>

                </actions>

                <events>

                    <event>

                        <id>id</id>

                        <name>name</name>

                        <type>start</type>

                    </event>

                </events>

                <forks>

                    <fork>

                        <condition>condition</condition>

                        <id>id</id>

                    </fork>

                </forks>

                <track-name>name</track-name>

            </arg0>

        </ns2:addTrack>

    </S:Body>

</S:Envelope>



Рисунок 6 - Добавление новой сущности по идентификатору.

Удаление сущности по идентификатору

<?xml version="1.0" encoding="UTF-8"?><S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">

    <SOAP-ENV:Header/>

    <S:Body>

        <ns2:deleteTrack xmlns:ns2="http://services.lab7.mycompany.com/">

            <arg0>111</arg0>

        </ns2:deleteTrack>

    </S:Body>

</S:Envelope>



РИсунок 7 - Удаление сущности по идентификатору.

1. Клиент представляет собой обычное консольное приложение с возможностью отправки запросов на веб сервис. Запуск клиента представлен на рисунках 8-9.



Рисунок 8 - Получение всех сущностей.



Рисунок 9 - Получение сущности по идентификатору.

ВЫВОД

В ходе выполнения лабораторной работы был разработан сессионный компонент без состояния с представлением в виде веб-сервиса и создан клиент для веб-сервиса.