

Санкт-Петербургский *Национальный Исследовательский*
Университет ИТМО
Факультет программной инженерии и компьютерной техники

Методы и средства программной инженерии
Лабораторная работа №3
Вариант 1423

Выполнил:
студент группы Р3231
Нестеров Иван Алексеевич

Преподаватель:
Цопа Евгений Алексеевич

г. Санкт-Петербург
2022 г.

Цель работы:

Написать сценарий для утилиты Apache Ant, реализующий компиляцию, тестирование и упаковку в jar-архив кода проекта из лабораторной работы №2 по дисциплине "Веб-программирование".

Каждый этап должен быть выделен в отдельный блок сценария; все переменные и константы, используемые в сценарии, должны быть вынесены в отдельный файл параметров; MANIFEST.MF должен содержать информацию о версии и о запускаемом классе.

Ход работы:

По ходу работы было в рабочую директорию был продублирован код лабораторной работы №2 по веб-программированию и написал build.xml сценарий сборки приложения. Содержимое build.xml:

```
<?xml version="1.0"?>
<project name="Basics-of-SE-Lab-3" default="report">

    <property name="src" location="src"/>
    <property name="class" location="target/classes"/>
    <property name="jar" location="target/artifacts"/>
    <property name="reports" location="target/reports"/>
    <property name="main-class"
location="src/main/java/com/lab2/servlets/ControllerServlet.java"/>

    <path id="classpath">
        <fileset dir="lib" includes="**/*.jar"/>
    </path>

    <target name="compile" description="Compiling project sources">
        <echo>Compiling project sources...</echo>
        <mkdir dir="${class}"/>
        <javac includeantruntime="true"
            srcdir="${src}"
            destdir="${class}">

            <classpath>
                <path refid="classpath"/>
            </classpath>
        </javac>
        <echo>Compilation completed successfully!</echo>
    </target>

    <target name="build" depends="compile" description="Packaging compiled
files into an executable jar archive">
        <echo>Building...</echo>
        <mkdir dir="${jar}"/>
        <jar basedir="${class}" destfile="${jar}/${ant.project.name}.jar">
            <manifest>
                <attribute name="Main-Class" value="${main-class}"/>
                <attribute name="Version" value="1.0"/>
            </manifest>
        </jar>
        <echo>Packaging completed successfully!</echo>
    </target>
```

```

    <target name="clean" description="Deleting compiled project classes and
all temporary files">
        <echo>Cleaning...</echo>
        <delete dir="${class}" />
        <echo>Cleaning completed successfully!</echo>
    </target>

    <target name="test" depends="build" description="Running project jUnit-
tests">
        <echo>Program testing...</echo>
        <junit printsummary="true" haltonfailure="true">
            <classpath>
                <path refid="classpath" />
                <pathelement location="${class}" />
                <pathelement location="${main-class}" />
                <pathelement location="${class}" />
            </classpath>
            <test name="ResultCheckerTest" todir="target/reports" />
            <formatter type="plain" usefile="false" />
        </junit>
        <echo>Program testing completed successfully!</echo>
    </target>

    <target name="xml" description="Validation of all xml files in the
project">
        <echo>XML project files validation...</echo>
        <xmlvalidate lenient="true">
            <fileset dir="." includes="**/*.xml">
                <exclude name=".idea/**" />
            </fileset>
        </xmlvalidate>
        <echo>XML files validation completed successfully!</echo>
    </target>

    <target name="report" depends="test">
        <echo>Report creating...</echo>
        <junit printsummary="true" haltonfailure="true">
            <classpath>
                <path refid="classpath" />
                <pathelement location="${class}" />
                <pathelement location="${main-class}" />
                <pathelement location="${class}" />
            </classpath>
            <test name="ResultCheckerTest" todir="target/reports" />
            <formatter type="xml" usefile="true" />
        </junit>
        <exec executable="git">
            <arg value="add" />
            <arg value="${reports}/*" />
        </exec>
        <input message="Please enter commit message:"
            addproperty="msg" />
        <exec executable="git">
            <arg value="commit" />
            <arg value="-m" />
            <arg value="${msg}" />
        </exec>
        <echo>Report creating completed successfully!</echo>
    </target>

</project>

```

Вывод: в ходе работы я ознакомился с модульным тестированием программ на языке Java, основными jUnit API, а также ознакомился с системами сборки, а при помощи одной из них, Ant, выполнил поставленные в лабораторной работе цели.