XProgram webservice quick setup

1. Install Wildfly (JBoss)

Use following instruction to install and configure Wildfly (JBoss) 17.0.1.Final on Ubuntu 18.04 LTS: https://vitux.com/install-and-configure-wildfly-jboss-on-ubuntu/

2. Deploy spring-resteasy.war to the server

Following instructions contains information how to deploy applications on WildFly using the Web Console and the CLI: http://www.mastertheboss.com/jboss-server/jboss-deploy/deploying-applications-on-wildfly-using-the-web-console-and-the-cli

3. Configure program to execute

Copy *holidays.cbl* and *xholidays.cbl* into **/vagrant/cobol/webservices/xholidays** folder on the server and compile them:

```
cd /vagrant/cobol/webservices/xholidays
. /opt/cobol-it4-64/bin/cobol-it-setup.sh
cobc -b xholidays.cbl holidays.cbl
```

In the **/vagrant/cobol/webservices/xholidays** folder create **run.sh** script with following content:

```
#!/bin/bash
export COBOLIT_LICENSE=/opt/cobol-it4-64/citlicense.xml
COBOLITDIR=/opt/cobol-it4-64
PATH=$COBOLITDIR/bin:${PATH}
LD_LIBRARY_PATH="$COBOLITDIR/lib:${LD_LIBRARY_PATH:=}"
DYLD_LIBRARY_PATH="$COBOLITDIR/lib:${DYLD_LIBRARY_PATH:=}"
SHLIB_PATH="$COBOLITDIR/lib:${SHLIB_PATH:=}"
LIBPATH="$COBOLITDIR/lib:${LIBPATH:=}"
COB="COBOL-IT"
COB_ERROR_FILE=/tmp/coberrplus
export COB_FILE_PATH=/tmp
export COB COBOLITDIR LD_LIBRARY_PATH PATH DYLD_LIBRARY_PATH SHLIB_PATH
LIBPATH COB_ERROR_FILE
cobcrun xholidays
```

4. Make POST requests

Use cURL to make a POST request to http://localhost:8080/spring-resteasy/xprogram:

```
curl -X POST -H "Content-Type: text/plain" -d @getstring
http://localhost:8081/spring-resteasy/xprogram
```

Implementation details

All source codes are located in spring-resteasy.zip, README.html contains information how to rebuild application war.

Following quickstart example is used as initial project:
https://github.com/wildfly/quickstart/tree/17.0.1.Final/spring-resteasy

Added XProgram bean, the bean is initialized with a program execution working directory and with path to a program to execute. The process method otf the bean accepts a string as argument, executes configured command line program and returns program execution output:
src/main/java/org/jboss/as/quickstarts/resteasyspring/XProgramBean.java

```
package org.jboss.as.quickstarts.resteasyspring;

import java.io.BufferedReader;
import java.io.File;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.PrintWriter;
import java.util.function.Consumer;

public class XProgramBean {

    private String workDirPath;
    private String programToRun;
    private boolean isWindows;

    public XProgramBean(String workDirPath, String programToRun) {
        this.workDirPath = workDirPath;
        this.programToRun = programToRun;
        this.isWindows =
System.getProperty("os.name").toLowerCase().startsWith("windows");
    }

    public String process(String request) {
        StringBuilder result = new StringBuilder();
        try {
            ProcessBuilder builder = new ProcessBuilder();
            if (isWindows) {
                builder.command("cmd.exe", "/c", programToRun);
            } else {
                builder.command("sh", "-c", programToRun);
            }
            builder.directory(new File(workDirPath));

            System.out.println("Work directory: " + workDirPath);
            System.out.println("Command: " + String.join(" ",
builder.command()));
```

```
            Process process = builder.start();
            PrintWriter pw = new PrintWriter(process.getOutputStream());
            pw.write(request);
            pw.write("\n");
            pw.flush();

            StreamGobbler streamGobbler = new
StreamGobbler(process.getInputStream(), str -> {
                System.out.println(str);
                result.append(str).append("\n");
            });
            streamGobbler.run();

            int exitCode = process.waitFor();
            System.out.println("Exit code: " + exitCode);
        } catch (Exception e) {
            e.printStackTrace();
        }
        return result.toString();
    }

    private static class StreamGobbler {
        private InputStream inputStream;
        private Consumer<String> consumer;

        public StreamGobbler(InputStream inputStream, Consumer<String>
consumer) {
            this.inputStream = inputStream;
            this.consumer = consumer;
        }

        public void run() {
            new BufferedReader(new
InputStreamReader(inputStream)).lines().forEach(consumer);
        }
    }
}
```

Added ability to make POST requests to the XProgram bean:
src/main/java/org/jboss/as/quickstarts/resteasyspring/XProgramSpringResource.java

```
package org.jboss.as.quickstarts.resteasyspring;

import javax.ws.rs.Consumes;
import javax.ws.rs.FormParam;
import javax.ws.rs.core.MediaType;
import javax.ws.rs.core.Response;
import javax.ws.rs.POST;
import javax.ws.rs.Path;
```

```
import javax.ws.rs.Produces;

import org.springframework.beans.factory.annotation.Autowired;

@Path("/")
public class XProgramSpringResource {

    @Autowired
    XProgramBean xProgramBean;

    @POST
    @Path("xprogram")
    @Consumes("text/plain")
    @Produces("text/plain")
    public Response postXProgram(String body) {
        String result = xProgramBean.process(body);
        return Response.ok(result).build();
    }

    @POST
    @Path("/xprogram-form")
    @Consumes(MediaType.APPLICATION_FORM_URLENCODED)
    public Response postXProgramForm(@FormParam("msg") String msg) {
        String result = xProgramBean.process(msg);
        return Response.ok(result).build();
    }
}
```

New beans added to application context: **src/main/webapp/WEB-INF/applicationContext.xml**

```
    <!-- XProgram bean -->
    <bean id="xProgramBean"
class="org.jboss.as.quickstarts.resteasyspring.XProgramBean">
        <!-- Program execution working directory -->
        <constructor-arg index="0" type="java.lang.String"
value="/vagrant/cobol/webservices/xholidays" />
        <!-- Program to execute -->
        <constructor-arg index="1" type="java.lang.String"
value="/vagrant/cobol/webservices/xholidays/run.sh" />
    </bean>

    <!-- JAX-RS XProgram resource -->
    <bean id="xProgramSpringResource"
class="org.jboss.as.quickstarts.resteasyspring.XProgramSpringResource" />
```