

2- Configurando AppVeyor

- Crear una cuenta y evaluar que pasa en cada una de las herramientas . Usar el login con github para entrar a la primera.
- Configuramos el proyecto de Maven


MSBUILD **SCRIPT** OFF

Build script


PS PS CORE **CMD** SH OFF

```
cd trabajo-practico-04/java-docker
mvn clean package
```

Save

 AppVeyor

PROJECTS ENVIRONMENTS DOCS SUPPORT

 {{SESSION.USER().FULLNAME}}

```
2.2.0.jar
733 Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/maven-project/2.2.0/maven-project-2.2.0.jar (153 KB
at 335.8 KB/sec)
734 Downloading: https://repo.maven.apache.org/maven2/org/vafer/jdependency/0.7/jdependency-0.7.jar
735 Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/maven-profile/2.2.0/maven-profile-2.2.0.jar (35 KB
at 73.6 KB/sec)
736 Downloading: https://repo.maven.apache.org/maven2/commons-io/commons-io/1.3.2/commons-io-1.3.2.jar
737 Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/maven-artifact-manager/2.2.0/maven-artifact-
manager-2.2.0.jar (64 KB at 136.1 KB/sec)
738 Downloading: https://repo.maven.apache.org/maven2/asm/asm-analysis/3.2/asm-analysis-3.2.jar
739 Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/maven-plugin-registry/2.2.0/maven-plugin-registry-
2.2.0.jar (30 KB at 56.2 KB/sec)
740 Downloading: https://repo.maven.apache.org/maven2/asm/asm-util/3.2/asm-util-3.2.jar
741 Downloaded: https://repo.maven.apache.org/maven2/org/vafer/jdependency/0.7/jdependency-0.7.jar (12 KB at 22.3 KB/sec)
742 Downloaded: https://repo.maven.apache.org/maven2/asm/asm-analysis/3.2/asm-analysis-3.2.jar (18 KB at 34.0 KB/sec)
743 Downloaded: https://repo.maven.apache.org/maven2/commons-io/commons-io/1.3.2/commons-io-1.3.2.jar (86 KB at 161.1
KB/sec)
744 Downloaded: https://repo.maven.apache.org/maven2/backport-util-concurrent/backport-util-concurrent/3.1/backport-util-
concurrent-3.1.jar (324 KB at 608.9 KB/sec)
745 Downloaded: https://repo.maven.apache.org/maven2/asm/asm-util/3.2/asm-util-3.2.jar (36 KB at 65.3 KB/sec)
746 [INFO] Including ch.qos.logback:logback-classic:jar:1.2.1 in the shaded jar.
747 [INFO] Including ch.qos.logback:logback-core:jar:1.2.1 in the shaded jar.
748 [INFO] Including org.slf4j:slf4j-api:jar:1.7.22 in the shaded jar.
749 [INFO] Minimizing jar ar.edu.ucc:ejemplo:jar:1.0-SNAPSHOT
750 [INFO] Minimized 617 -> 421 (68%)
751 [INFO] Replacing C:\projects\ing-software-3\trabajo-practico-04\java-docker\target\ejemplo.jar with C:\projects\ing-
software-3\trabajo-practico-04\java-docker\target\ejemplo-1.0-SNAPSHOT-shaded.jar
752 [INFO]
753 [INFO] BUILD SUCCESS
754 [INFO]
755 [INFO] Total time: 24.836 s
756 [INFO] Finished at: 2018-09-17T22:14:01+00:00
757 [INFO] Final Memory: 20M/48M
758 [INFO]
759 Discovering tests...OK
760 Build success
```

Artifacts

PATH TO ARTIFACT

DEPLOYMENT NAME

TYPE

trabajo-practico-04\java-docker\target\ejemp

SalidaBuild

Auto

Add artifact

Save

```
AppVeyor PROJECTS ENVIRONMENTS DOCS SUPPORT [SESSION.USER().FULLNAME]
tree-2.0.jar (55 KB at 120.8 KB/sec)
736 Downloading: https://repo.maven.apache.org/maven2/commons-io/commons-io/1.3.2/commons-io-1.3.2.jar
737 Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/maven-project/2.2.0/maven-project-2.2.0.jar (153 KB at 325.1 KB/sec)
738 Downloading: https://repo.maven.apache.org/maven2/asm/asm-analysis/3.2/asm-analysis-3.2.jar
739 Downloaded: https://repo.maven.apache.org/maven2/backport-util-concurrent/backport-util-concurrent/3.1/backport-util-concurrent-3.1.jar (324 KB at 690.7 KB/sec)
740 Downloading: https://repo.maven.apache.org/maven2/asm/asm-util/3.2/asm-util-3.2.jar
741 Downloaded: https://repo.maven.apache.org/maven2/org/vafer/jdependency/0.7/jdependency-0.7.jar (12 KB at 23.8 KB/sec)
742 Downloaded: https://repo.maven.apache.org/maven2/org/apache/maven/maven-plugin-registry/2.2.0/maven-plugin-registry-2.2.0.jar (30 KB at 59.8 KB/sec)
743 Downloaded: https://repo.maven.apache.org/maven2/asm/asm-util/3.2/asm-util-3.2.jar (36 KB at 71.4 KB/sec)
744 Downloaded: https://repo.maven.apache.org/maven2/commons-io/commons-io/1.3.2/commons-io-1.3.2.jar (86 KB at 171.4 KB/sec)
745 Downloaded: https://repo.maven.apache.org/maven2/asm/asm-analysis/3.2/asm-analysis-3.2.jar (18 KB at 35.1 KB/sec)
746 [INFO] Including ch.qos.logback:logback-classic:jar:1.2.1 in the shaded jar.
747 [INFO] Including ch.qos.logback:logback-core:jar:1.2.1 in the shaded jar.
748 [INFO] Including org.slf4j:slf4j-api:jar:1.7.22 in the shaded jar.
749 [INFO] Minimizing jar ar.edu.ucc:ejemplo:jar:1.0-SNAPSHOT
750 [INFO] Minimized 617 -> 421 (68%)
751 [INFO] Replacing C:\projects\ing-software-3\trabajo-practico-04\java-docker\target\ejemplo.jar with C:\projects\ing-software-3\trabajo-practico-04\java-docker\target\ejemplo-1.0-SNAPSHOT-shaded.jar
752 [INFO]
753 [INFO] BUILD SUCCESS
754 [INFO]
755 [INFO] Total time: 19.113 s
756 [INFO] Finished at: 2018-09-17T22:26:12+00:00
757 [INFO] Final Memory: 19M/47M
758 [INFO]
759 Discovering tests...OK
760 Collecting artifacts...
761 Found artifact 'trabajo-practico-04\java-docker\target\ejemplo.jar' matching 'trabajo-practico-04\java-docker\target\ejemplo.jar' path
762 Uploading artifacts...
763 [1/1] trabajo-practico-04\java-docker\target\ejemplo.jar (559,637 bytes)...100%
764 Build success
```

Acá podemos generar los builds y publicarlos

Para ejecutarlo: **java -jar ejemplo.jar**

```
[MacBook-Pro-de-Laura:Downloads laurapognante$ java -jar ejemplo.jar
19:31:35.722 [main] INFO ar.edu.ucc.App - Hola Mundo!
```

LATEST BUILD HISTORY DEPLOYMENTS EVENTS SETTINGS

General

Environment

Build

Tests

Artifacts

Deployment

NuGet

Notifications

Permissions

Badges

Export YAML

Delete project

Export project configuration to YAML

Below is `appveyor.yml` for your current project configuration. To store versioned project configuration along with your sources download `appveyor.yml` and commit it to the root of your code repository.

Download appveyor.yml

```
version: 1.0.{build}
build_script:
- cmd: >-
  cd trabajo-practico-04/java-docker
  mvn clean package
artifacts:
- path: trabajo-practico-04\java-docker\target\ejemplo.jar
  name: SalidaBuild
```


PORQUE UN YAML???

Secreto:

- En todo momento como lo tengo en git bajo control de configuración, se como aplique la configuración y como la construí. Si necesito volver 6 meses atrás y saber que use para construirlo, puedo volver y hacerlo.

- Si viene alguien nuevo y quiere usarlo, con esto hereda del mio la configuración. Seteando en mi propia tool, y no tiene que hacer nada más que correrlo, haciendo solamente el build.
- Cuando lo pongo en mi repo ya sabe cómo construirse

Optional:

LATEST BUILD	HISTORY	DEPLOYMENTS	EVENTS	SETTINGS
General	Project status badge			
Environment	Current status			
Build				
Tests	SVG image URL			
Artifacts	<input type="text" value="https://ci.appveyor.com/api/projects/status/10aobeoajvwnlapw?svg=true"/>			
Deployment	Raster image URL			
NuGet	<input type="text" value="https://ci.appveyor.com/api/projects/status/10aobeoajvwnlapw"/>			
Notifications	Sample markdown code			
Permissions	<input type="text" value="[![Build status](https://ci.appveyor.com/api/projects/status/10aobeoajvwnlapw?svg=true)](https://ci.appveyor.com/project/lalipognante/ing-software-3)"/>			
Badges	master branch status badge			
Export YAML	SVG branch image URL			
Delete project	<input type="text" value="https://ci.appveyor.com/api/projects/status/10aobeoajvwnlapw/branch/master"/>			

si pongo en mi repo:

```
[![Build status](https://ci.appveyor.com/api/projects/status/10aobeoajvwnlapw?svg=true)](https://ci.appveyor.com/project/lalipognante/ing-software-3)]
```

- Con un **BADGES**: Le voy a hacer saber a la gente que mi build y mi código complica.

Cuando entro al README voy a saber muchas cosas de mi repo. Por ejemplo que el build está corriendo.