




AVENTURA TEXTUAL

Nombre: Joseba Hernando Moisés

Asignatura: Diseño y Mantenimiento del software



Índice de contenido

-ENUNCIADO DEL PROYECTO	3
-REQUISITOS.....	3
-FORMATO ARCHIVO XML	4
-ESTRUCTURA DTD.....	4
-XML DE EJEMPLO.....	5
-ESQUEMA PARA ENTENDERLO MEJOR EL XML	7
-EJEMPLO DE LA EJECUCION	8
-DIAGRAMAS.....	8
-SCENE	8
-WORLD.....	8
-PROP	9
-PLAYER	9
-INTERACTION.....	9
-MAIN.....	9
-TODOS.....	10
-PATRON	11
-PROBLEMAS ENCONTRADOS	11
-POSIBLES MEJORAS QUE SE PUEDEN HACER	11
-FUENTES.....	11
-Construir tdd se indica la estructura que debe cumplir el XML.....	11
-Parsear documento xml	11
-Construccion tdt	12
-DocumentBuilderFactory	12

-ENUNCIADO DEL PROYECTO

El proyecto consistirá en **completar el motor de ficción interactiva** comenzado en los ejercicios en clase (enunciado y una posible solución), a realizar **individualmente**. El lenguaje de implementación será **Java**.

Se realizará **una única entrega** a través de un **repositorio online** (GitHub), el día **24 de enero** como muy tarde, con un peso del **40%** sobre la nota final de la asignatura.

-REQUISITOS

1. **Como evaluador quiero acceso al código y su historial para ver la implementación.**
Existe un commit final en github.
2. **Como evaluador quiero una memoria para evaluar el uso de buenas prácticas y patrones de diseño**
He realizado este informe con índice.
3. **Como evaluador quiero instrucciones para poder construir y lanzar la aplicación.**
He realizado un video tutorial disponible en onedrive.
4. **Como cliente quiero que la historia pueda tener varias escenas.**
El XML dispone de 4 escenas.
5. **Como cliente quiero poder interactuar con elementos de la escena.**
Podemos abrir cajas y movernos por las puertas.
6. **Como cliente quiero poder moverme entre escenas.**
A través de ciertas puertas podemos ir de una escena a otra.
7. **Como cliente quiero que las interacciones con un elemento puedan afectar a otros elementos.**
Podemos abrir cajas y cambiar el estado de las puertas dependiendo de si tenemos llave.
8. **Como cliente quiero poder crear mis propias historias, definiéndolas en fichero.**
He definido un .tdt donde se indica la estructura que debe cumplir el XML.

-FORMATO ARCHIVO XML

world

player [scene]

scenes

scene [id, numDoors, numBoxes, msj]

action [dest, msj] (dest es el prop destinatario de la accion, es decir, el prop que se ejecuta al elegir esa opcion)

props

prop [id, type, state, scene01, scene02] (caso door: type puede ser door o box, scene01 y scene02 indican las dos escenas que une esa puerta)

action [condition, msj] (caso door: condition puede ser open o closed)

prop [id, type, state, scene01, unlock] (caso box: type puede ser door o box, scene01 indica la escena a la que pertenece el prop y unlock es el prop que desbloquea al ejecutarse)

action [msj] (caso box)

-ESTRUCTURA DTD

<!ELEMENT world (player,scenes,props)>

<!ELEMENT player EMPTY>

<!ATTLIST player
scene CDATA #REQUIRED>

<!ELEMENT scenes (scene)+>

<!ELEMENT props (prop)*>

<!ELEMENT scene (action)*>

<!ATTLIST scene
id ID #REQUIRED
numDoors CDATA #REQUIRED
numBoxes CDATA #REQUIRED
msj CDATA #REQUIRED>

<!ELEMENT prop (action)+>

<!ATTLIST prop
id ID #REQUIRED
type CDATA #REQUIRED
state CDATA #IMPLIED
scene01 CDATA #REQUIRED
scene02 CDATA #IMPLIED
unlock CDATA #IMPLIED>

<!ELEMENT action EMPTY>

<!ATTLIST action

condition CDATA #IMPLIED
dest CDATA #IMPLIED
msj CDATA #REQUIRED>

-XML DE EJEMPLO

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE world SYSTEM "example.dtd">
<world>
  <player scene="scene00"/>
  <scenes>
    <scene id="scene00" numDoors="3" numBoxes="1" msj="Te encuentras en
una habitación oscura con solo una caja en el centro y tres puertas (una al este, una al oeste y
otra al sur).">
      <action dest="box00" msj="Abrir la caja"/>
      <action dest="door00" msj="Ir hacia el este"/>
      <action dest="door01" msj="Ir hacia el oeste"/>
      <action dest="door02" msj="Ir hacia el sur"/>
    </scene>
    <scene id="scene01" numDoors="3" numBoxes="2" msj="En esta habitación
hay dos cajas y tres puertas (una al este, una al oeste y otra al sur).">
      <action dest="box01" msj="Abrir la primera caja"/>
      <action dest="box02" msj="Abrir la segunda caja"/>
      <action dest="door03" msj="Ir hacia el este"/>
      <action dest="door00" msj="Ir hacia el oeste"/>
      <action dest="door04" msj="Ir hacia el sur"/>
    </scene>
    <scene id="scene02" numDoors="3" numBoxes="1" msj="En esta habitación se
respira mejor, solo hay una caja en el centro y tres puertas (una al norte, una al este y otra al
oeste).">
      <action dest="box03" msj="Abrir la caja"/>
      <action dest="door02" msj="Ir hacia el norte"/>
      <action dest="door05" msj="Ir hacia el este"/>
      <action dest="door06" msj="Ir hacia el oeste"/>
    </scene>
    <scene id="scene03" numDoors="2" numBoxes="1" msj="Te encuentras en
una habitación con el ambiente enrarecido, una caja en el centro y dos puertas (una al norte y
otra al oeste).">
      <action dest="box04" msj="Abrir la caja"/>
      <action dest="door04" msj="Ir hacia el norte"/>
      <action dest="door05" msj="Ir hacia el oeste"/>
    </scene>
    <scene id="final" numDoors="0" numBoxes="0" msj="Final del juego, has
ganado. Enhorabuena!!"/>
  </scenes>
  <props>
    <prop id="door00" type="door" state="false" scene01="scene00"
scene02="scene01">
      <action condition="open" msj="La puerta se encuentra abierta y la
cruzamos."/>
    </prop>
  </props>
</world>
```

```

        <action condition="closed" msj="La puerta se encuentra cerrada, es
necesaria una llave."/>
    </prop>

```

```

    <prop id="door01" type="door" state="true" scene01="scene00"
scene02="scene00">
        <action condition="open" msj="La puerta se encuentra abierta, pero
detrás hay un muro."/>
        <action condition="closed" msj="La puerta se encuentra cerrada, es
necesaria una llave."/>
    </prop>

```

```

    <prop id="door02" type="door" state="false" scene01="scene00"
scene02="scene02">
        <action condition="open" msj="La puerta se encuentra abierta y la
cruzamos."/>
        <action condition="closed" msj="La puerta se encuentra cerrada, es
necesaria una llave."/>
    </prop>

```

```

    <prop id="door03" type="door" state="true" scene01="scene01"
scene02="scene01">
        <action condition="open" msj="La puerta se encuentra abierta, pero
detrás hay un muro."/>
        <action condition="closed" msj="La puerta se encuentra cerrada, es
necesaria una llave."/>
    </prop>

```

```

    <prop id="door04" type="door" state="true" scene01="scene01"
scene02="scene03">
        <action condition="open" msj="La puerta se encuentra abierta y la
cruzamos."/>
        <action condition="closed" msj="La puerta se encuentra cerrada, es
necesaria una llave."/>
    </prop>

```

```

    <prop id="door05" type="door" state="false" scene01="scene03"
scene02="scene02">
        <action condition="open" msj="La puerta se encuentra abierta y la
cruzamos."/>
        <action condition="closed" msj="La puerta se encuentra cerrada y no
tiene cerradura, que raro..."/>
    </prop>

```

```

    <prop id="door06" type="door" state="false" scene01="scene02"
scene02="final">
        <action condition="open" msj="La puerta se encuentra abierta y la
cruzamos, hemos conseguido salir al exterior!!"/>
        <action condition="closed" msj="La puerta se encuentra cerrada y
parece diferente..."/>
    </prop>

```

```

<prop id="box00" type="box" scene01="scene00" unlock="door00">
  <action msj="La caja se encuentra abierta y dentro hay una llave con el
mensaje 'puerta 0' grabado."/>
</prop>

<prop id="box01" type="box" scene01="scene01" unlock="null">
  <action msj="La caja se encuentra vacía."/>
</prop>

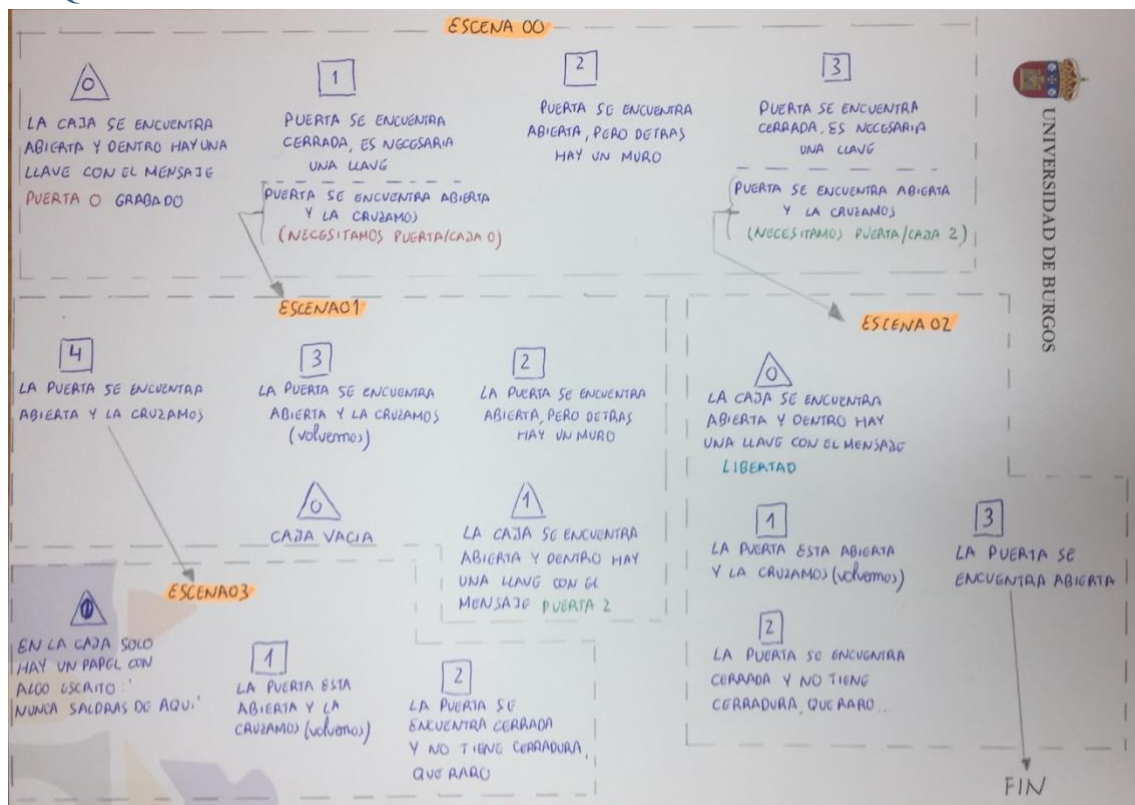
<prop id="box02" type="box" scene01="scene01" unlock="door02">
  <action msj="La caja se encuentra abierta y dentro hay una llave con el
mensaje 'puerta 2' grabado."/>
</prop>

<prop id="box03" type="box" scene01="scene02" unlock="door06">
  <action msj="La caja se encuentra abierta y dentro hay una llave con el
mensaje 'libertad' grabado."/>
</prop>

<prop id="box04" type="box" scene01="scene03" unlock="null">
  <action msj="En la caja solo hay un papel con algo escrito: 'NUNCA
SALDRÁS DE AQUÍ'."/>
</prop>
</props>
</world>

```

-ESQUEMA PARA ENTENDERLO MEJOR EL XML



-EJEMPLO DE LA EJECUCION

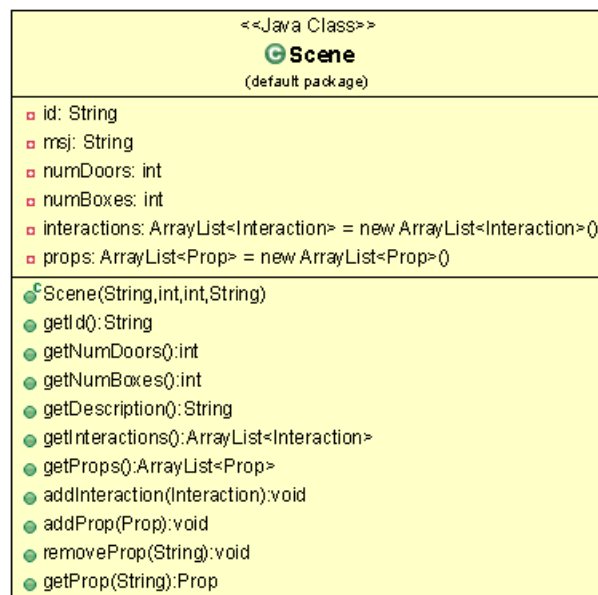
-Adjunto video subido a **onedrive** con permisos de **lectura** y **ejecución**. (recomiendo vlc para reproducir correctamente ya que en el navegador a veces dependiendo cual utilices)

[https://universidaddeburgos-](https://universidaddeburgos-my.sharepoint.com/:v/g/personal/jhm0025_alu_ubu_es/EUcPcL6v171LvyfQugNVhnIB4D56h0u36i4cwKXOrjix_w?e=8xS8cL)

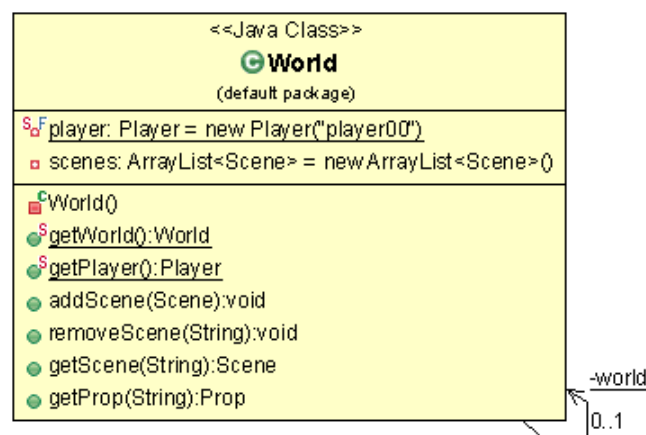
my.sharepoint.com/:v/g/personal/jhm0025_alu_ubu_es/EUcPcL6v171LvyfQugNVhnIB4D56h0u36i4cwKXOrjix_w?e=8xS8cL

-DIAGRAMAS

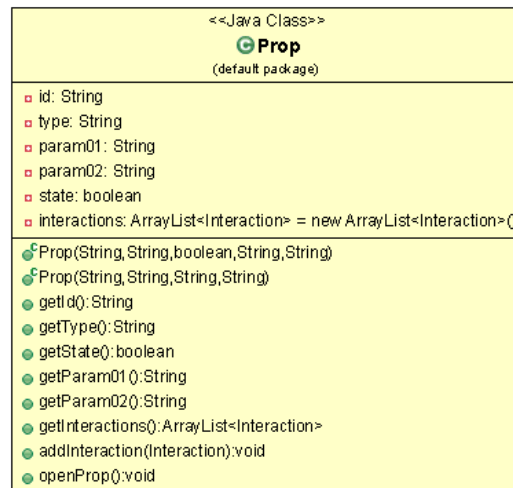
-SCENE



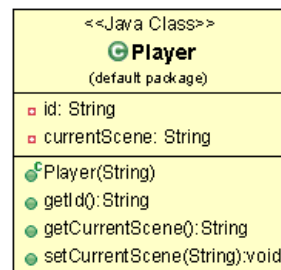
-WORLD



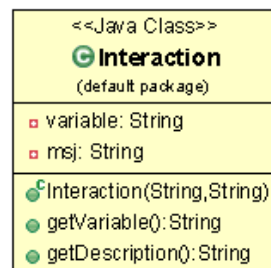
-PROP



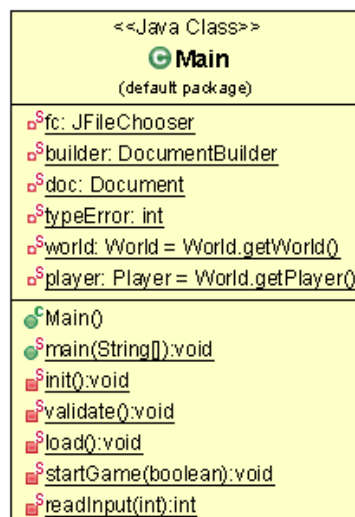
-PLAYER



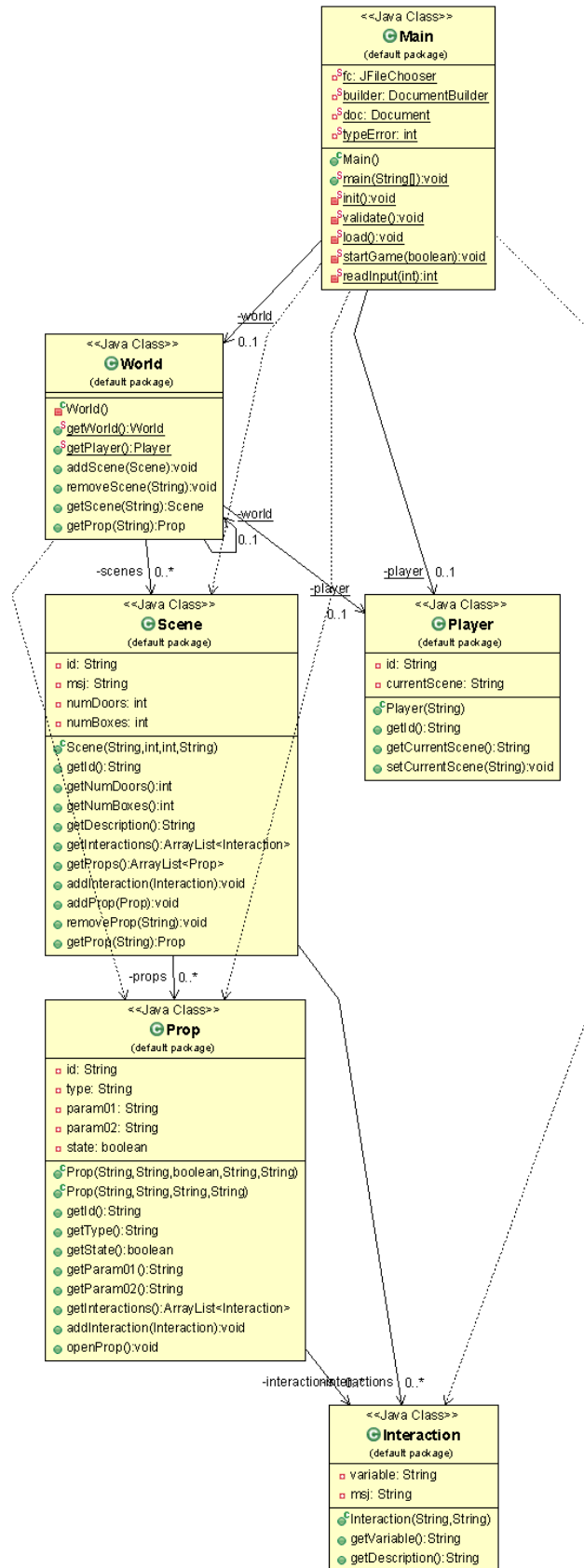
-INTERACTION



-MAIN



-TODOS



-PATRON

He usado el patrón ***singleton*** para que se garantice la unicidad de ***World***, solo existiendo una instancia. Recordamos que la intención de este patrón es garantizar que una clase sólo tenga una instancia y proporcionar un punto de acceso global a ella.

-PROBLEMAS ENCONTRADOS

Para cargar el XML una opción era poner un path fijo, al final conseguí realizar algo más dinámico que es que al ejecutar la app aparece una ventana para seleccionar el XML que queremos cargar.

Crear el ***dtd*** también me ha quitado bastante tiempo, pero lo veo necesario para cumplir el requisito 8 de poder crear propias historias y seguir la misma estructura.

Lo que más tiempo me ha llevado de la practica fue el ***parseo*** y la ***correcta lectura del XML***. Para ello tuve que recordar apuntes de la asignatura de Procesadores del Lenguaje en el que hicimos una práctica de parseo.

A modo de pequeña conclusión decir que la posible solución que subiste me ha servido de mucho, ayudándome sobre todo en generar la estructura global de la práctica y los métodos. También decir que me ha llevado muchas, muchas horas realizar la práctica correctamente (sobre todo lectura del XML).

-POSIBLES MEJORAS QUE SE PUEDEN HACER

- Podríamos hacer que las llaves abrieran cajas y no solo puertas.
- Usuario tenga vidas que vayan disminuyendo si caen a puertas con un muro.
- Añadir más objetos en las escenas como por ejemplo lámparas para encender la luz de una habitación que se encuentre oscura.

-FUENTES

-Construir tdd se indica la estructura que debe cumplir el XML.

<https://jorgesanchez.net/manuales/xml/xml-validacion.html>

<https://www.youtube.com/watch?v=fPU1ex7bSgg>

-Parsear documento xml

https://www.tutorialspoint.com/java_xml/java_dom_parse_document.htm

Apuntes Cesar Osorio de la asignatura de Procesadores del lenguaje

-Construccion tdt

<http://www.mundolinux.info/sintaxis-de-las-dtd.htm>

-DocumentBuilderFactory

<https://docs.oracle.com/javase/8/docs/api/javax/xml/parsers/DocumentBuilderFactory.html>

<https://renidev.javabolivia.com/2011/05/30/simple-dom-documentbuilder-para-java/>

FIN