

DOCUMENTACIÓN ENTREGABLE 1

Arquitectura del Software

Grupo: Trivial 1A
Sergio Cueto López de Bustamante
Ignacio Rodríguez Vázquez
Yenni Ramos Martínez
Diego Jaular Ortigueira
Robert Stefanita Ene
Alejandro García Torriello
Isabel Del Álamo Rancaño
Francisco Gil Gala

Contenido

Planteamiento del Problema	3
Requisitos Funcionales.....	3
Requisitos No Funcionales	4
Identificación de los Interesados (Stakeholders)	4
Atributos de Calidad	5
Atributos de calidad e Interesados	5
Escenarios de Calidad	6
Vistas.....	7
Contexto.....	7
Summary	8
Details	9
Logging	13
Summary	13
Details	14
Persistencia	17
Summary	17
Details	18
Vista Procesado.....	21
Summary	21
Details	22
Stereotypes.....	26
Manual de usuario	31
1. Introducción	31
2. ¿Cómo jugar?	32
Manual del sistema.....	33
Instalación y ejecución de la aplicación:.....	33
Instalación de del entorno Eclipse:	34
Instalación del plugin "M2Eclipse":	34
Instalación y ejecución de la base de datos:	35
Modificar el path de windows:	35

Este documento incluye:

- Una identificación de stakeholders, atributos y escenarios de calidad para el problema propuesto, junto con la relación que guardan.
- La arquitectura optada que soluciona los requisitos pedidos, explicada y detallada.
- El manual de usuario, indicando lo necesario para que se pueda manejar la aplicación.
- Un manual de sistema, indicando aspectos importantes de configuración y restricciones para el correcto funcionamiento de la misma

Planteamiento del Problema

La empresa NoGame dedicada a la creación de videojuegos quiere crear una variante del juego Trivial, aunque la idea es que puedan crear más juegos similares en el futuro.

Inicialmente se creará una base de datos MongoDB con preguntas suficientes para poder abastecer a los diferentes juegos. Para ello, construirán una aplicación que lea ficheros con las preguntas y respuestas, procese dichas preguntas indicando si hay errores o no y las almacene en la base de datos.

Los ficheros se leerán de bancos de preguntas, con formato GIFT, ya existentes. La aplicación utilizará una representación interna de las preguntas en formato JSON lo cual facilita su posterior almacenamiento en la base de datos.

La aplicación se ejecutará en dos etapas. Una primera etapa analizará los ficheros con las preguntas y generará el formato JSON; y otra etapa tomará las preguntas en JSON y las almacenará en la base de datos MongoDB. La ejecución de estas 2 etapas estará controlada por un operario de la compañía que podrá decidir cuándo ejecuta cada etapa e incluso podrá automatizar dicha ejecución para que se realice cada cierto tiempo.

La compañía no requiere que las aplicaciones sean muy eficientes ni que el proceso de conversión se realice de forma interactiva. Aunque en esta primera fase, la compañía solamente requiere la conversión de preguntas en formato GIFT, se está pensando que la solución debe admitir otros formatos en el futuro.

Requisitos Funcionales

1. Se le debe poder comunicar a la aplicación por consola de donde tiene que leer la información.
2. Se le debe poder comunicar a la aplicación por consola donde ha de devolver el resultado.
3. La aplicación debe poder leer principalmente ficheros en un formato determinado (GIFT) con la información de las preguntas , y generar a partir de él otro fichero en un formato intermedio (JSON)
4. La aplicación debe permitir observar los resultados intermedios de la conversión para detectar errores.
5. La aplicación debe poder leer los ficheros en el formato intermedio (JSON) y almacenarlos en una base de datos.
6. Estas dos operaciones podrán ser realizadas individualmente por un operario
7. Estas dos operaciones se deben poder automatizar, programando su ejecución cada cierto tiempo.

Requisitos No Funcionales

1. Debe poder correr en paralelo varias instancias de la aplicación a la vez.
2. Se le debe poder comunicar a la aplicación por consola que formato va a leer.

Identificación de los Interesados (Stakeholders)

- **Responsables de NoGame:** Se trata de los equipos directivos de la empresa, son responsables de los presupuestos y toman las decisiones que comprometen fondos de dicho presupuesto.
- **Equipo de desarrollo del proyecto Trivial:** Este equipo será el responsable de desarrollar el sistema resultante de la arquitectura.
- **Responsables de la información:** Este equipo se encargará de introducir la información con las preguntas que utilizarán los distintos juegos, además de comprobar su veracidad y actualizarlas en caso de cambios.

Código	Stakeholders	Intereses
ST-01	Responsables de NoGame	<ol style="list-style-type: none">1. Bajo coste económico de desarrollo. El desarrollo del proyecto debe ser corto y con un coste reducido.2. Posibilidad de que el juego sea multiplataforma.3. Posibilidad de crear más juegos con formato preguntas/respuestas, extendiendo el que se está diseñando actualmente.
ST-02	Equipo de desarrollo del proyecto Trivial	<ol style="list-style-type: none">1. Baja complejidad en los algoritmos de aleatorización de las preguntas.
ST-03	Responsables de la información	<ol style="list-style-type: none">2. Proyecto rentable, esto es, que permita ser desarrollado por el precio establecido con un grado de rentabilidad que haga atractivo el desarrollo.3. Permitir la ampliación de la aplicación con nuevas partes de manera sencilla al proyecto.1. Máxima facilidad y rapidez a la hora de introducir preguntas, preferiblemente con tratamientos por lotes y no individualmente.2. Máxima facilidad y rapidez para buscar una determinada pregunta.3. Máxima facilidad y rapidez para actualizar los datos de una pregunta.4. Poder mantener la integridad y coherencia en la información.

Atributos de Calidad

Los diferentes atributos de calidad son de interés para alguno de los Stakeholders. La siguiente tabla muestra la lista de intereses para el proyecto actual:

Código	Descripción	Tipo de Atributo
AT001	El sistema debe estar disponible siempre que el operario lo quiera usar.	Disponibilidad
AT002	Facilidad de cambio de los algoritmos de conversión, ya que puede ser necesario modificarlos en el futuro para admitir más formatos de entrada, o ampliar la variedad de formatos de salida.	Modificabilidad
AT003	Facilidad para probar la fiabilidad de sistema, garantía de que las conversiones se han realizado correctamente.	Testabilidad
AT004	Es importante la seguridad en la información procesada. Los datos que se guarden son muy importantes para el correcto funcionamiento de la aplicación final.	Seguridad
AT005	Es importante poder saber si se ha realizado la conversión de un formato a otro de forma correcta, y si se ha almacenado correctamente.	Testabilidad
AT006	No exige una interfaz gráfica, pero sí que el operador que la use pueda, mediante comandos, realizar una serie de operaciones básicas. Interacción con el usuario.	Usabilidad

Atributos de calidad e Interesados

Los diferentes atributos de calidad son de interés para alguno de los Stakeholders. La siguiente tabla muestra la lista de intereses para el proyecto actual:

Atributos Vs Interesados	ST-01	ST-02	ST-03
AT001		X	X
AT002		X	X
AT003		X	X
AT004	X	X	X
AT005		X	X

AT006	X	X
-------	---	---

Escenarios de Calidad

Con toda la información anterior se procederá a definir los escenarios de calidad que influyen en esta arquitectura.

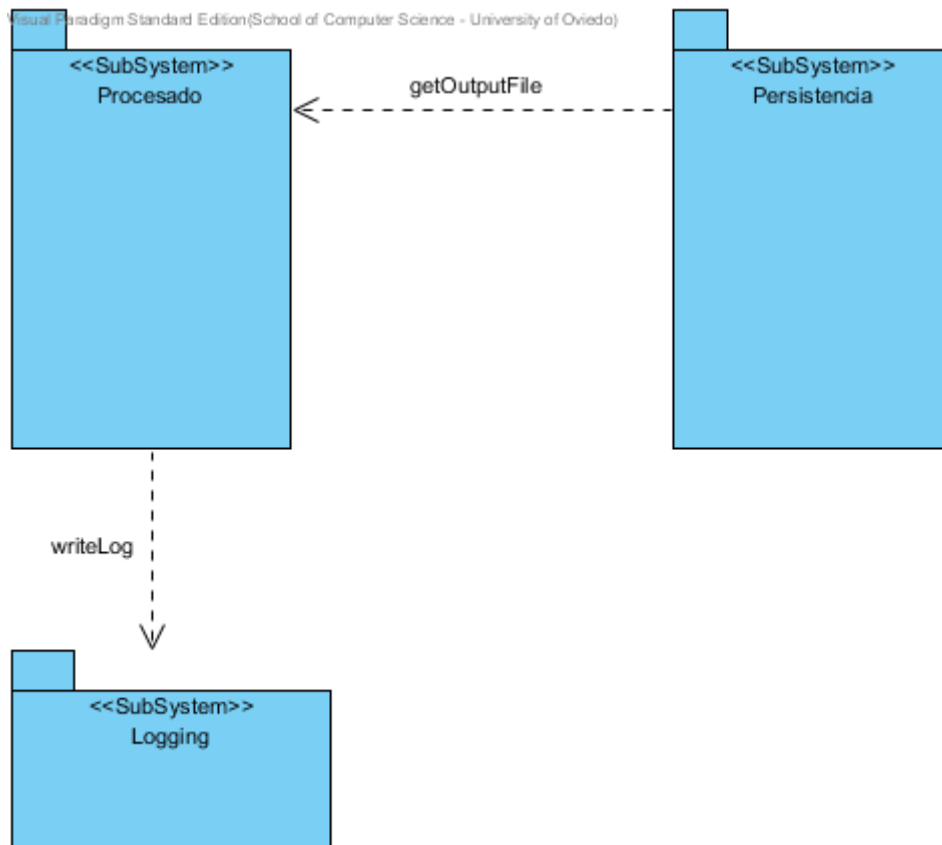
Escenario	Fuente de estímulo	de Entorno	Artefacto	Respuesta	Medición de la respuesta	Atributos de calidad afectado
1	Modificaciones y ampliación del proyecto	Sistema en periodo de desarrollo	Código fuente	Facilidad para modificar o ampliar proyecto	Tiempo empleado en realizar el cambio	AT002
2	Corrección de errores	Sistema en Fase de desarrollo	Código fuente	Facilidad para la detección y corrección de errores	Minimizar los errores en el menor tiempo posible	AT002 AT005 AT003
3	Usar el sistema en cualquier momento	Explotación	Sistema	Disponibilidad y usabilidad del sistema cuando el usuario lo requiere	Satisfacción del usuario	AT001 AT006
4	Acceso a las respuestas de las preguntas	Explotación	Datos internos y base de datos	Restricción del acceso a datos almacenados	Minimizar el acceso a información privada del sistema	AT004

Vistas

Vistas	Stakeholders	Atributos de calidad	Escenarios
Logging	ST-02, ST-03	AT002, AT005, AT003	2
Persistencia	ST-01, ST-02, ST-03	AT004, AT002, AT005, AT003	2, 4
Vista procesado	ST-02, ST-03	AT001, AT006, AT002, AT005, AT003	1, 2, 3




Package Diagram

Contexto



Name	Value
Name	Contexto
Author	grupo 1 A
Create Date Time	19-feb-2015 9:42:27
Last Modified	05-mar-2015 10:18:23
Shape Presentation Option	0

Summary

Name	Documentation
 Procesado	Es un sistema que realiza las siguientes operaciones: lee de un fichero en un formato (inicialmente GIFT), lo transforma a un formato intermedio , y después de ese formato obtiene un fichero en el formato de salida (JSON en este caso).
 Persistencia	Almacena un fichero en un determinado formato en una base de datos. Se comunica con el subsistema de procesamiento para obtener los datos a guardar.
 Logging	subsistema que permite comprobar el proceso de conversión de preguntas.

Documentation

En esta primera fase se obtiene una aplicación que permitirá a un operario guardar preguntas y respuestas al repositorio de información de la aplicación.

Esta información tendrá un formato de entrada, GIFT en una primera instancia, que tendrá que ser analizada y transformada en un formato intermedio, JSON , para que después sea almacenada.

El operario debe poder elegir entre hacer las tareas de lectura y guardado independientemente, o si realizarlas de forma secuencial.

Además, podrá ver información relevante sobre el procesado de las preguntas aportadas en el fichero indicado.

La comunicación del usuario con la aplicación se lleva a cabo mediante una consola de comandos, que tiene la capacidad de traducir las órdenes que introducen los usuarios, mediante un conjunto de instrucciones facilitadas por él mismo directamente al núcleo y al conjunto de herramientas que forman la aplicación. Las órdenes se introducen siguiendo la sintaxis incorporada por dicho intérprete, Al ingresar la orden con la tecla 'Intro', el intérprete analiza la secuencia de caracteres ingresada y, si la sintaxis de la orden es correcta, la ejecuta, recurriendo para ello a las funciones que ofrece la aplicación. La respuesta al usuario se representa en el monitor o en forma de segundo plano. Se trabaja de manera interactiva, es decir, usuario y máquina se comunican de forma sucesiva.

El estilo de desarrollo sigue un estilo Pipe & Filter ,en el que la información es procesada mediante filtros, como el parser y el serializer, y en el que la información de salida de uno es la

que va a tratar el siguiente. Este estilo tiene como ventaja la reusabilidad de los filtros, ya la ampliabilidad de los mismo, permitiendo a la aplicación leer multitud de formatos cambiando únicamente un filtro.

Details



Procesado



Name	Value	
Documentation	Es un sistema que realiza las siguientes operaciones: lee de un fichero en un formato (inicialmente GIFT), lo transforma a un formato intermedio , y después de ese formato obtiene un fichero en el formato de salida (JSON en este caso).	
Abstract	false	
Leaf	false	
Root	false	
Stereotypes	SubSystem	
Visibility	public	
Project Management	Name	Value
	Author	grupo 1 A
	Create Date Time	05-feb-2015 9:46:36
	Last Modified	03-mar-2015 23:24:32

Children


Name	Documentation
Input File Parser	<p>componente que será el que se encargue de recibir el fichero de entrada y procesarlo, devolviendo un objeto intermedio que será postprocesado después. Inicialmente será un parser de formato .GIFT.</p> <p>Al existir este componente así separado, se podrá modificar el tipo de fichero que se va a recibir, ampliando así la gama de formatos a procesar.</p>
Outpuf File Serializer	<p>el serializador obtiene el objeto intermedio del preprocesado. Es con la información que contiene este con la que generará el fichero de salida, en formato JSON en este caso. Al ser independiente del preprocesado, con cambiar este elemento permitiremos que el operario pueda , a partir de un fichero en un</p>

	mismo formato, obtener ficheros en formatos distintos.
--	--

Relationships

writeLog : Dependency		
To	 Logging	
Visibility	Unspecified	
Project Management	Name	Value
	Author	Nacho
	Create Date Time	03-mar-2015 22:17:41
	Last Modified	03-mar-2015 23:14:32
getOutputFile : Dependency		
From	 Persistencia	
Visibility	Unspecified	
Project Management	Name	Value
	Author	Nacho
	Create Date Time	03-mar-2015 22:01:26
	Last Modified	03-mar-2015 23:14:32

Sub Diagrams

Name	Documentation
 Vista Procesado	Es un sistema que realiza las siguientes operaciones: lee de un fichero en un formato (inicialmente GIFT), lo transforma a un formato intermedio , y después de ese formato obtiene un fichero en el formato de salida (JSON en este caso).






Persistencia


Name	Value
Documentation	Almacena un fichero en un determinado formato en una base de datos. Se comunica con el subsistema de procesamiento para obtener los datos a guardar.
Abstract	false
Leaf	false

Root	false	
Stereotypes	SubSystem	
Visibility	public	
Project Management	Name	Value
	Author	grupo 1 A
	Create Date Time	05-feb-2015 9:48:47
	Last Modified	03-mar-2015 23:24:32


Children

Name	Documentation
 persistor	Este subsistema se encarga de guardar ,a petición del usuario o de forma automatizada, los ficheros obtenidos tras el procesamiento en una base de datos.
 getOutputFile	Necesita comunicarse con el sistema de procesado para obtener la información que debe guardar. De esta forma obtiene un fichero ,previamente procesado, y en un formato intermedio , JSON en nuestro caso, que será el que guarde.
 save	permite comunicarse con el operario ,para que este pueda guardar la información que ha obtenido en el momento que quiera, pudiendo asi dividir el proceso en dos fases.

Relationships

getOutputFile : Dependency		
To	 Procesado	
Visibility	Unspecified	
Project Management	Name	Value
	Author	Nacho
	Create Date Time	03-mar-2015 22:01:26
	Last Modified	03-mar-2015 23:14:32

Sub Diagrams




Name	Documentation
 Persistencia	Almacena un fichero en un determinado formato en una base de datos. Se comunica con el

	subsistema de procesamiento para obtener los datos a guardar.
--	---


Logging

Name	Value	
Documentation	subsistema que permite comprobar el proceso de conversión de preguntas.	
Abstract	false	
Leaf	false	
Root	false	
Stereotypes	SubSystem	
Visibility	public	
Project Management	Name	Value
	Author	uo231201
	Create Date Time	05-feb-2015 9:48:47
	Last Modified	03-mar-2015 23:14:32

Children


Name	Documentation
 Logger	componente del sistema de logging que se comunica con el sistema de procesamiento , permitiendo que este guarde la información del procesado.
 writeLog	recibe la petición del sistema de procesado para guardar en el log como ha ido el procesado de las preguntas, y los posibles errores que hayan surgido.
 getLog	Permite al usuario obtener la información relativa al procesado de las preguntas.

Relationships

writeLog : Dependency		
From	 Procesado	
Visibility	Unspecified	
Project Management	Name	Value
	Author	Nacho
	Create Date Time	03-mar-2015 22:17:41

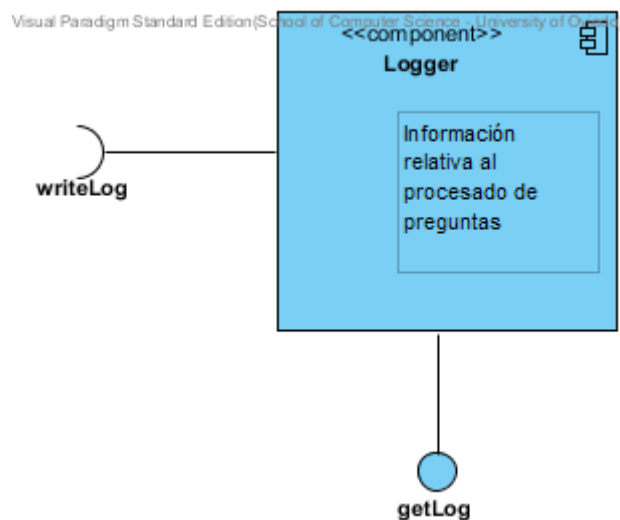
	Last Modified	03-mar-2015 23:14:32
--	---------------	----------------------

Sub Diagrams

Name	Documentation
 Logging	subsistema que permite comprobar el proceso de conversión de preguntas.



Component Diagram


Logging



Name	Value
Name	Logging
Author	grupo 1 A
Create Date Time	03-mar-2015 22:18:00
Last Modified	03-mar-2015 23:41:18
Shape Presentation Option	0

Summary

Name	Documentation
 Logger	componente del sistema de logging que se comunica con el sistema de procesamiento, permitiendo que este guarde la información del procesado.
 writeLog	recibe la petición del sistema de procesamiento para guardar en el log como ha ido el procesamiento de las

	preguntas, y los posibles errores que hayan surgido.
 getLog	Permite al usuario obtener la información relativa al procesamiento de las preguntas.

Documentation

subsistema que permite comprobar el proceso de conversión de preguntas.


Details




Logger

Name	Value	
Documentation	componente del sistema de logging que se comunica con el sistema de procesamiento , permitiendo que este guarde la información del procesamiento.	
Active	false	
Business Key Mutable	true	
Business Model	false	
Visibility	public	
Abstract	false	
Leaf	false	
Root	false	
Indirectly Instantiated	true	
Project Management	Name	Value
	Author	grupo 1 A
	Create Date Time	03-mar-2015 22:22:02
	Last Modified	03-mar-2015 23:24:32

Relationships


Unnamed Usage		
To	 writeLog	
Visibility	Unspecified	
Project Management	Name	Value
	Author	Nacho
	Create Date Time	03-mar-2015 22:22:42
	Last Modified	03-mar-2015 22:23:06
Unnamed Realization		

From	 getLog	
Visibility	Unspecified	
Project Management	Name	Value
	Author	Nacho
	Create Date Time	03-mar-2015 22:22:52
	Last Modified	03-mar-2015 22:23:06

writeLog

Name	Value	
Documentation	recibe la petición del sistema de procesado para guardar en el log como ha ido el procesado de las preguntas, y los posibles errores que hayan surgido.	
Active	false	
Business Key Mutable	true	
Business Model	false	
Visibility	public	
Leaf	false	
Root	false	
Stereotypes	Interface	
Project Management	Name	Value
	Author	grupo 1 A
	Create Date Time	03-mar-2015 22:22:42
	Last Modified	03-mar-2015 23:24:32

Relationships

Unnamed Usage		
From	 Logger	
Visibility	Unspecified	
Project Management	Name	Value
	Author	Nacho
	Create Date Time	03-mar-2015 22:22:42
	Last Modified	03-mar-2015 22:23:06



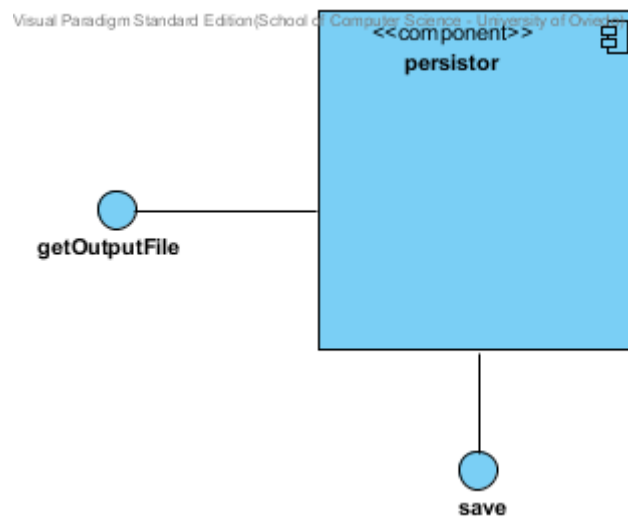
Name	Value	
Documentation	Permite al usuario obtener la información relativa al procesado de las preguntas.	
Active	false	
Business Key Mutable	true	
Business Model	false	
Visibility	public	
Leaf	false	
Root	false	
Stereotypes	Interface	
Project Management	Name	Value
	Author	grupo 1 A
	Create Date Time	03-mar-2015 22:22:52
	Last Modified	03-mar-2015 23:24:32

Relationships

Unnamed Realization		
To	Logger	
Visibility	Unspecified	
Project Management	Name	Value
	Author	Nacho
	Create Date Time	03-mar-2015 22:22:52
	Last Modified	03-mar-2015 22:23:06




Component Diagram

Persistencia



Name	Value
Name	Persistencia
Author	grupo 1 A
Create Date Time	26-feb-2015 10:39:47
Last Modified	03-mar-2015 23:41:18
Shape Presentation Option	0

Summary

Name	Documentation
 persistor	Este subsistema se encarga de guardar ,a petición del usuario o de forma automatizada, los ficheros obtenidos tras el procesamiento en una base de datos.
 getOutputFile	Necesita comunicarse con el sistema de procesamiento para obtener la información que debe guardar. De esta forma obtiene un fichero ,previamente procesado, y en un formato intermedio , JSON en nuestro caso, que será el que guarde.
 save	permite comunicarse con el operario ,para que este pueda guardar la información que ha obtenido en el momento que quiera, pudiendo así dividir el proceso en dos fases.

Documentation


Almacena un fichero en un determinado formato en una base de datos. Se comunica con el subsistema de procesamiento para obtener los datos a guardar.


Details



Name	Value	
Documentation	Este subsistema se encarga de guardar ,a petición del usuario o de forma automatizada, los ficheros obtenidos tras el procesamiento en una base de datos.	
Active	false	
Business Key Mutable	true	
Business Model	false	
Visibility	public	
Abstract	false	
Leaf	false	
Root	false	
Indirectly Instantiated	true	
Project Management	Name	Value
	Author	grupo 1 A
	Create Date Time	26-feb-2015 10:53:58
	Last Modified	03-mar-2015 23:24:32

Relationships

Unnamed Realization		
From	 getOutputFile	
Visibility	Unspecified	
Project Management	Name	Value
	Author	uo231201
	Create Date Time	26-feb-2015 10:58:16
	Last Modified	26-feb-2015 11:00:40

Unnamed Realization		
From	 save	
Visibility	Unspecified	
Project Management	Name	Value
	Author	Nacho
	Create Date Time	03-mar-2015 22:18:38

	Last Modified	03-mar-2015 22:23:06
--	---------------	----------------------



getOutputFile

Name	Value	
Documentation	Necesita comunicarse con el sistema de procesamiento para obtener la información que debe guardar. De esta forma obtiene un fichero ,previamente procesado, y en un formato intermedio , JSON en nuestro caso, que será el que guarde.	
Active	false	
Business Key Mutable	true	
Business Model	false	
Visibility	public	
Leaf	false	
Root	false	
Stereotypes	Interface	
Project Management	Name	Value
	Author	grupo 1 A
	Create Date Time	26-feb-2015 10:58:16
	Last Modified	03-mar-2015 23:24:32

Relationships

Unnamed Realization		
To	persistor	
Visibility	Unspecified	
Project Management	Name	Value
	Author	uo231201
	Create Date Time	26-feb-2015 10:58:16
	Last Modified	26-feb-2015 11:00:40




save

Name	Value
Documentation	permite comunicarse con el operario ,para que este pueda guardar la información que ha obtenido en el momento que quiera, pudiendo asi dividir el proceso en dos fases.

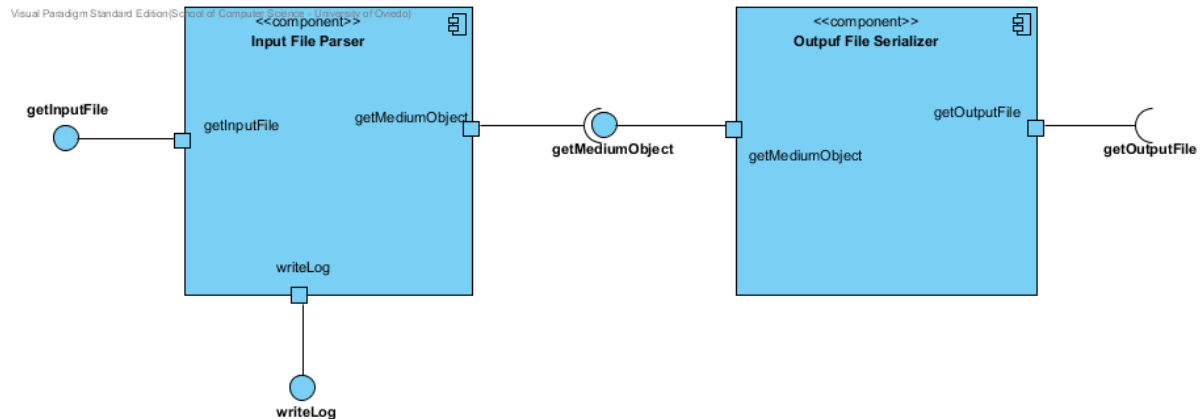
Active	false	
Business Key Mutable	true	
Business Model	false	
Visibility	public	
Leaf	false	
Root	false	
Stereotypes	Interface	
Project Management	Name	Value
	Author	grupo 1 A
	Create Date Time	03-mar-2015 22:18:38
	Last Modified	03-mar-2015 23:24:32

Relationships

Unnamed Realization		
To	 persistor	
Visibility	Unspecified	
Project Management	Name	Value
	Author	Nacho
	Create Date Time	03-mar-2015 22:18:38
	Last Modified	03-mar-2015 22:23:06





Component Diagram



Vista Procesado



Name	Value
Name	Vista Procesado
Author	grupo 1 A
Create Date Time	05-feb-2015 10:16:56
Last Modified	03-mar-2015 23:41:18
Shape Presentation Option	0

Summary

Name	Documentation
 Input File Parser	<p>componente que será el que se encargue de recibir el fichero de entrada y procesarlo, devolviendo un objeto intermedio que será postprocesado después. Inicialmente será un parser de formato .GIFT.</p> <p>Al existir este componente así separado, se podrá modificar el tipo de fichero que se va a recibir, ampliando así la gama de formatos a procesar.</p>
 Outpuf File Serializer	<p>el serializador obtiene el objeto intermedio del preprocesado. Es con la información que contiene este con la que generará el fichero de salida, en formato JSON en este caso. Al ser independiente del preprocesado, con cambiar este elemento permitiremos que el operario pueda, a partir de un fichero en un mismo formato, obtener ficheros en formatos distintos.</p>
 getMediumObject	<p>el serializador obtiene el objeto intermedio del preprocesado para tratarlo posteriormente.</p>
 getOutputFile	<p>permitirá a este sistema comunicarse con el sistema de persistencia, obteniendo este la información en un formato de salida, JSON inicialmente, para</p>

	después guardarla.
 getInputFile	permite al operario indicar cual será el fichero a procesar por el parser. Inicialmente solo procesará ficheros .GIFT .
 writeLog	permite guardar la información relativa al procesamiento de las preguntas en un sistema de logging independiente. El operario podrá acceder a este para comprobar como se desarrolló.

Documentation

Es un sistema que realiza las siguientes operaciones: lee de un fichero en un formato (inicialmente GIFT), lo transforma a un formato intermedio , y después de ese formato obtiene un fichero en el formato de salida (JSON en este caso).



Details



Input File Parser

Name	Value	
Documentation	<p>componente que será el que se encargue de recibir el fichero de entrada y procesarlo, devolviendo un objeto intermedio que será postprocesado después. Inicialmente será un parser de formato .GIFT.</p> <p>Al existir este componente así separado, se podrá modificar el tipo de fichero que se va a recibir, ampliando así la gama de formatos a procesar.</p>	
Active	false	
Business Key Mutable	true	
Business Model	false	
Visibility	public	
Abstract	false	
Leaf	false	
Root	false	
Indirectly Instantiated	true	
Project Management	Name	Value
	Author	grupo 1 A
	Create Date Time	05-feb-2015 10:17:12
	Last Modified	03-mar-2015 23:24:32

Children

Name	Documentation
 getInputFile	permite al operario indicar cuál será el fichero a procesar por el parses. Inicialmente solo procesará ficheros .GIFT .
 writeLog	permite guardar la información relativa al procesado de las preguntas en un sistema de logging independiente. El operario podrá acceder a este para comprobar cómo se desarrolló.



Outpuf File Serializer

Name	Value	
Documentation	el serializador obtiene el objeto intermedio del preprocesado. Es con la información que contiene este con la que generará el fichero de salida, en formato JSON en este caso. Al ser independiente del preprocesado, con cambiar este elemento permitiremos que el operario pueda , a partir de un fichero en un mismo formato, obtener ficheros en formatos distintos.	
Active	false	
Business Key Mutable	true	
Business Model	false	
Visibility	public	
Abstract	false	
Leaf	false	
Root	false	
Indirectly Instantiated	true	
Project Management	Name	Value
	Author	grupo 1 A
	Create Date Time	05-feb-2015 10:23:27
	Last Modified	03-mar-2015 23:24:32

Children

Name	Documentation
------	---------------

<ul style="list-style-type: none"> getMediumObject 	el serializador obtiene el objeto intermedio del preprocesado para tratarlo posteriormente.
<ul style="list-style-type: none"> getOutputFile 	permitirá a este sistema comunicarse con el sistema de persistencia, obteniendo esta la información en un formato de salida, JSON inicialmente, para después guardarla.



getMediumObject

Name	Value	
Documentation	el serializador obtiene el objeto intermedio del preprocesado para tratarlo posteriormente.	
Active	false	
Business Key Mutable	true	
Business Model	false	
Visibility	public	
Leaf	false	
Root	false	
Stereotypes	Interface	
Project Management	Name	Value
	Author	grupo 1 A
	Create Date Time	05-feb-2015 10:47:05
	Last Modified	03-mar-2015 23:24:32



getOutputFile

Name	Value	
Documentation	permitirá a este sistema comunicarse con el sistema de persistencia, obteniendo esta la información en un formato de salida, JSON inicialmente, para después guardarla.	
Active	false	
Business Key Mutable	true	
Business Model	false	
Visibility	public	
Leaf	false	
Root	false	
Stereotypes	Interface	
Project Management	Name	Value

	Author	grupo 1 A
	Create Date Time	26-feb-2015 10:56:16
	Last Modified	03-mar-2015 23:24:32



getInputFile

Name	Value	
Documentation	permite al operario indicar cual será el fichero a procesar por el parses. Inicialmente solo procesará ficheros .GIFT .	
Active	false	
Business Key Mutable	true	
Business Model	false	
Visibility	public	
Leaf	false	
Root	false	
Stereotypes	Interface	
Project Management	Name	Value
	Author	grupo 1 A
	Create Date Time	05-feb-2015 10:21:30
	Last Modified	03-mar-2015 23:24:32



writeLog

Name	Value
Documentation	permite guardar la información relativa al procesado de las preguntas en un sistema de logging independiente. El operario podrá acceder a este para comprobar como se desarrolló.
Active	false
Business Key Mutable	true
Business Model	false
Visibility	public
Leaf	false
Root	false
Stereotypes	Interface

Stereotypes

UseCase		
Tagged Value Definitions	Level	
	Type	Enumeration
	Enumeration Values	Summary, User, Subfunction
	Project Management	Name
		Value
		Author
		uo231201
	Create Date Time	26-feb-2015
		10:07:45
	Last Modified	26-feb-2015
		10:09:13
	Complexity	
	Type	Enumeration
	Enumeration Values	Low, Medium, High
	Project Management	Name
		Value
		Author
		uo231201
	Create Date Time	26-feb-2015
		10:07:45
	Last Modified	26-feb-2015
		10:09:13
	Use Case Status	
	Type	Enumeration
	Enumeration Values	Name Only, Initial, Base, Complete, Deferred
	Project Management	Name
		Value
		Author
		uo231201
	Create Date Time	26-feb-2015
		10:07:45
	Last Modified	26-feb-2015
		10:09:13
	Implementation Status	
	Type	Enumeration
	Enumeration Values	Scheduled, Started, Partially Complete, Complete, Partially Deferred

	Project Management		
		Name	Value
		Author	uo231201
		Create Date Time	26-feb-2015 10:07:45
		Last Modified	26-feb-2015 10:09:13
	Preconditions		
	Type	Multi-line Text	
	Project Management		
		Name	Value
		Author	uo231201
		Create Date Time	26-feb-2015 10:07:45
		Last Modified	26-feb-2015 10:09:13
	Post-conditions		
	Type	Multi-line Text	
	Project Management		
		Name	Value
		Author	uo231201
		Create Date Time	26-feb-2015 10:07:45
		Last Modified	26-feb-2015 10:09:13
	Author		
	Type	Text	
	Project Management		
		Name	Value
		Author	uo231201
		Create Date Time	26-feb-2015 10:07:45
		Last Modified	26-feb-2015 10:09:13
	Assumptions		
	Type	Multi-line Text	

	Project Management	Name	Value
		Author	uo231201
		Create Date Time	26-feb-2015 10:07:45
		Last Modified	26-feb-2015 10:09:13

requirement			
Tagged Value Definitions	Text		
	Type	HTML	
	Project Management	Name	Value
		Author	uo231201
		Create Date Time	26-feb-2015 10:07:41
		Last Modified	26-feb-2015 10:09:13
	ID		
	Type	Text	
	Project Management	Name	Value
		Author	uo231201
		Create Date Time	26-feb-2015 10:07:41
		Last Modified	26-feb-2015 10:09:13
	source		
	Type	Text	
	Project Management	Name	Value
		Author	uo231201
		Create Date Time	26-feb-2015 10:07:41
		Last Modified	26-feb-2015 10:09:13
	kind		

	Type	Enumeration	
	Enumeration Values	Functional, Performance, Interface	
	Project Management	Name	Value
		Author	uo231201
		Create Date Time	26-feb-2015 10:07:41
		Last Modified	26-feb-2015 10:09:13
	verifyMethod		
	Type	Enumeration	
	Enumeration Values	Analysis, Demonstration, Inspection, Test	
	Project Management	Name	Value
		Author	uo231201
		Create Date Time	26-feb-2015 10:07:41
		Last Modified	26-feb-2015 10:09:13
	risk		
	Type	Enumeration	
	Enumeration Values	High, Medium, Low	
	Project Management	Name	Value
		Author	uo231201
		Create Date Time	26-feb-2015 10:07:41
		Last Modified	26-feb-2015 10:09:13
	status		
	Type	Enumeration	
	Enumeration Values	Proposed, Approved, Rejected, Deferred, Implemented, Mandatory, Obsolete	
	Project Management	Name	Value
		Author	uo231201

		Create Date Time	26-feb-2015 10:07:41
		Last Modified	26-feb-2015 10:09:13

Interface

use

SubSystem

access

Manual de usuario

1. Introducción

En este documento se van a explicar las distintas operaciones que un usuario podrá realizar con nuestra aplicación, así como la forma y la secuencia en la que debe realizarlas.

El usuario utilizará la aplicación por línea de comandos, permitiéndole hacer las opciones mostradas en la Figura 1.

```
Wellcome to Trivial Extractor

1 - Leer fichero
2 - Mostrar contenido del fichero
3 - Guardar preguntas en la base de datos
4 - Leer de la base de datos
5 - Borrar la base de datos
6 - ¡Jugar!
0 - Salir
```

Figura 1.Opciones trivial

Las opciones que se presentan son las siguientes:

1. **Leer fichero:** esta opción lee desde un fichero externo ,este fichero contiene las preguntas y respuestas del juego.
2. **Mostrar contenido del fichero:** lista el contenido del fichero mostrando por consola para cada pregunta sus posibles respuestas.
3. **Guardar preguntas en la base de datos:** guarda las preguntas en una base de datos.
4. **Leer de la base de datos:** leer desde una base de datos las preguntas y respuestas del juego.
5. **Borrar base de datos:** borra las preguntas y respuestas contenidas en la base de datos.
6. **Jugar:** permite comenzar a interaccionar con el juego.
7. **Salir:** finaliza la ejecución de la aplicación.

El usuario puede interaccionar con la aplicación escribiendo la opción que desee procesar y pulsando Enter .En la Figura 2 podemos ver un ejemplo de una interacción de un usuario que quiere leer un fichero externo.

```
1
Introduce el nombre del fichero:
questions.gift
Fichero cargado correctamente!
```

Figura 2.Ejemplo Leer de un fichero

2. ¿Cómo jugar?

Tras introducir el entorno de la aplicación, para que el usuario pueda comenzar a jugar debe seguir los siguientes pasos:

1. Cargar las preguntas y respuestas que se utilizaran en el juego. Para hacer esto el usuario puede escoger entre estas dos opciones:
 - a. Leer desde un fichero externo.
 - b. Leer desde la base de datos.

Estas dos opciones se muestran en la Figura 3. El usuario deberá seleccionar opción 1 o 4 y presionar Enter.

```
Wellcome to Trivial Extractor

1 - Leer fichero
2 - Mostrar contenido del fichero
3 - Guardar preguntas en la base de datos
4 - Leer de la base de datos
5 - Borrar la base de datos
6 - ¡Jugar!
0 - Salir
```

Figura 3. Leer preguntas

2. Una vez cargadas las preguntas el siguiente paso es ejecutar la opción ¡Jugar! (6), como se muestra en la figura 4.

```
6
¿En qué deporte se usa tiza?
1) En el tenis
2) En el críquet
3) En el billar
4) En el curling
```

Figura 4. A Jugar

3. El siguiente paso será responder a la pregunta que aparecerá por pantalla. En la Figura 5 se puede ver como se mostrará la pregunta

```
¿Cuántas manos tiene un caballo?
1) Dos
2) Una
3) Ninguna
4) Cuatro
2
```

Figura 5. Formato de preguntas

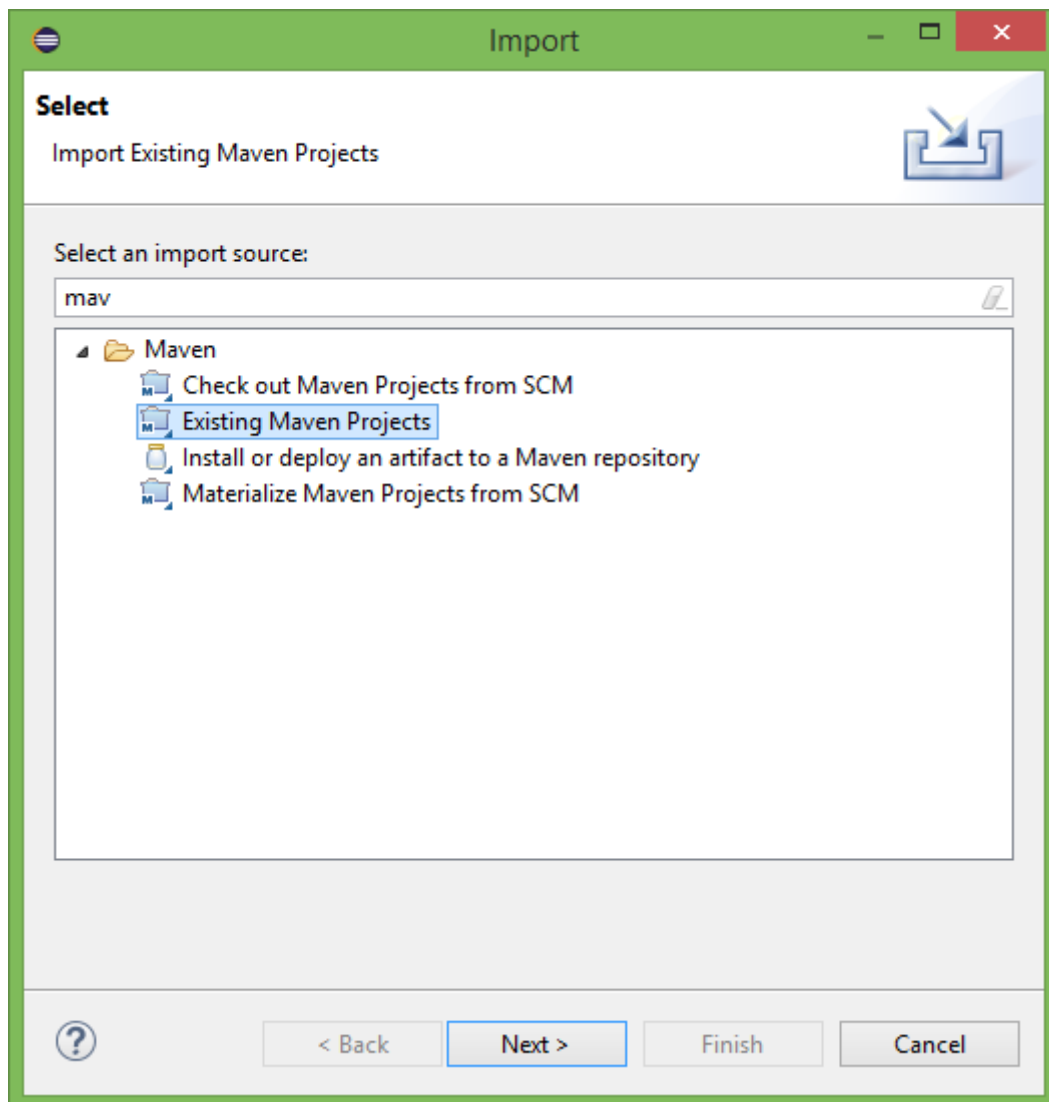
4. Para responder la pregunta el usuario solo tendrá que introducir el número de la opción que crea que es la correcta y presionar Enter. El juego le reportará si la respuesta era correcta o no.

Manual del sistema

Instalación y ejecución de la aplicación:

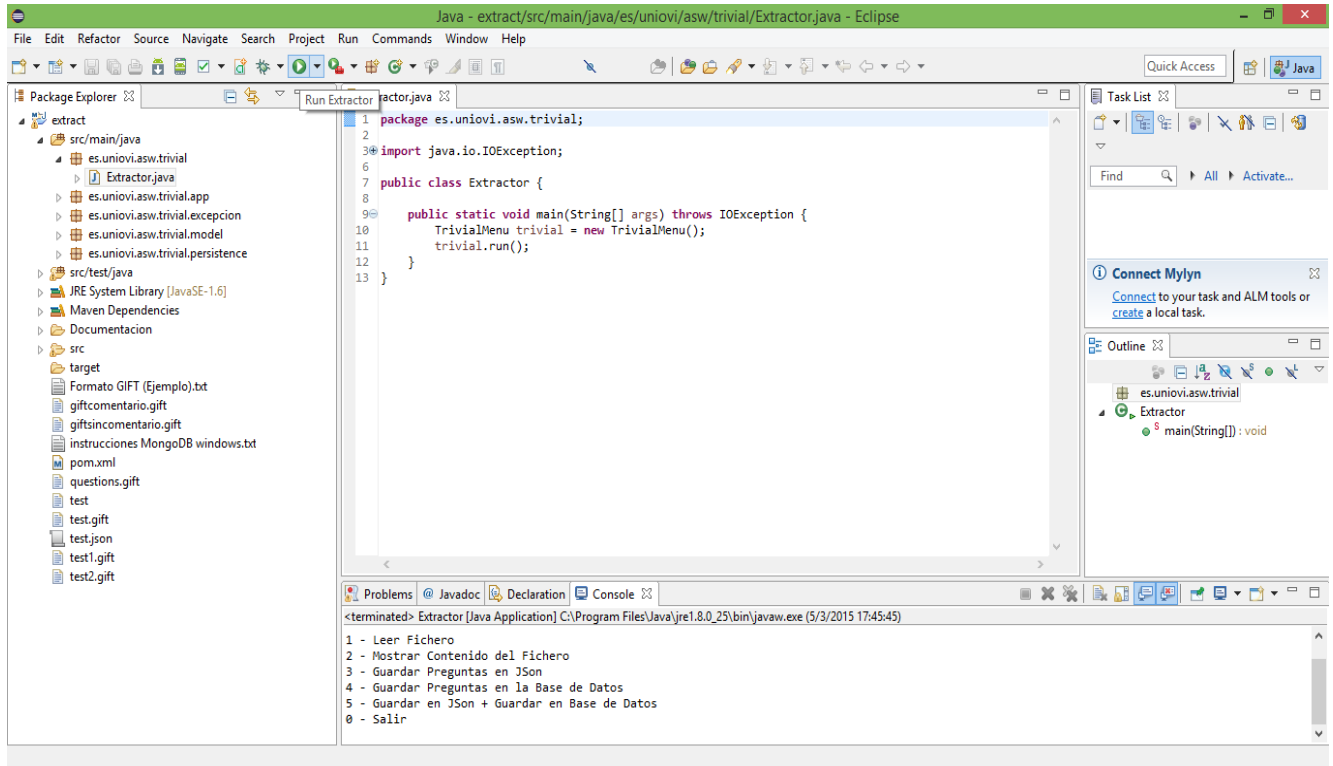
Por el momento la aplicación no tiene instalador, por lo que se debe ejecutar desde algún entorno de desarrollo como por ejemplo Eclipse.

Deberá importar el proyecto como un nuevo proyecto maven ([instalar el plugin M2Eclipse](#))



A continuación buscar la clase extractor, esta contiene el método main(),

Y darle a run:



Instalación de del entorno Eclipse:

En el caso de no disponer del entorno Eclipse deberá descargarlo desde el siguiente sitio:

<https://eclipse.org/>

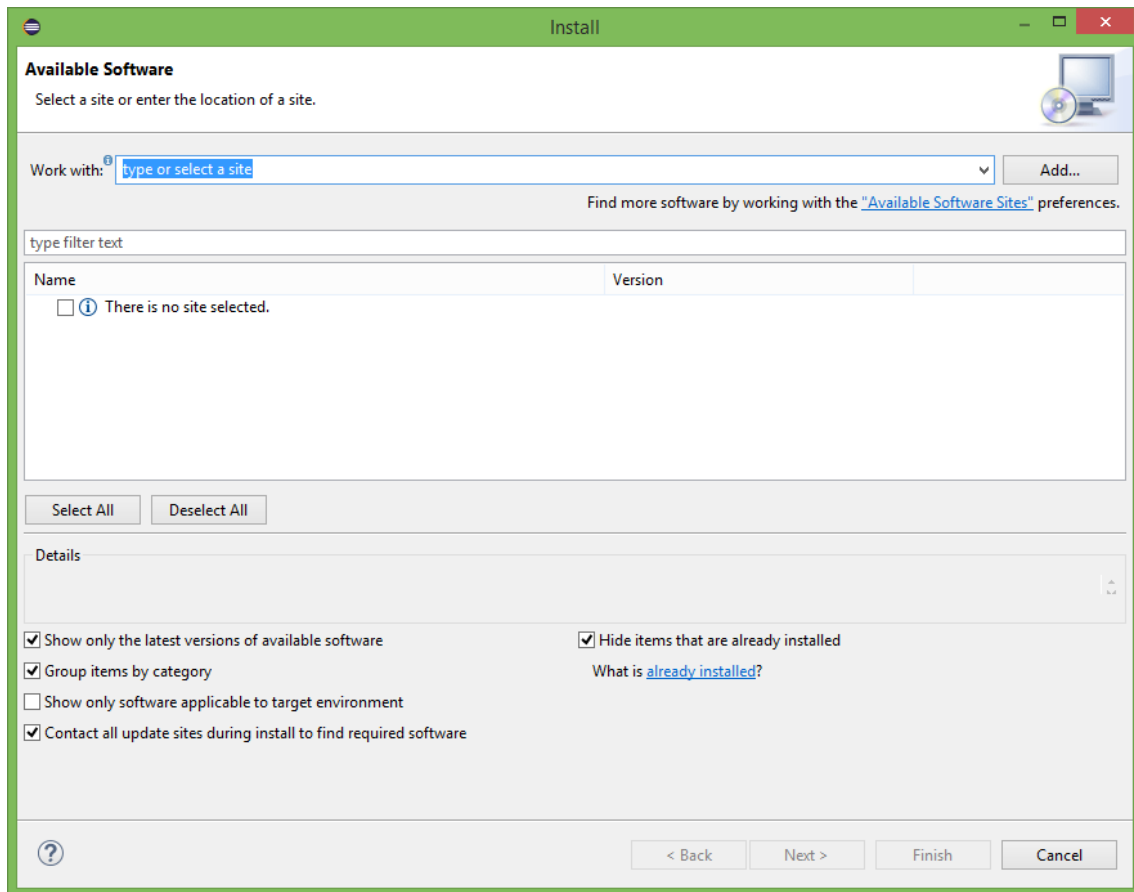
Y seguir las instrucciones que aparecerán en la pantalla.

Instalación del plugin “M2Eclipse”:

También deberá instalar el plugin de maven “M2Eclipse” en el caso de no tenerlo ya.

Abra el eclipse y valla a :

Help->instal new software...



Pulse en add y añada en name: Maven y en Location: ["http://download.eclipse.org/technology/m2e/releases"](http://download.eclipse.org/technology/m2e/releases)(sin las comillas)

Seleccione todas las casillas y acepte todo hasta finalizar la instalación.

Eclipse se reiniciará y ya tendrá el plugin instalado.

Instalación y ejecución de la base de datos:

Instrucciones para MongoDB 2.6.7 64-bit Windows

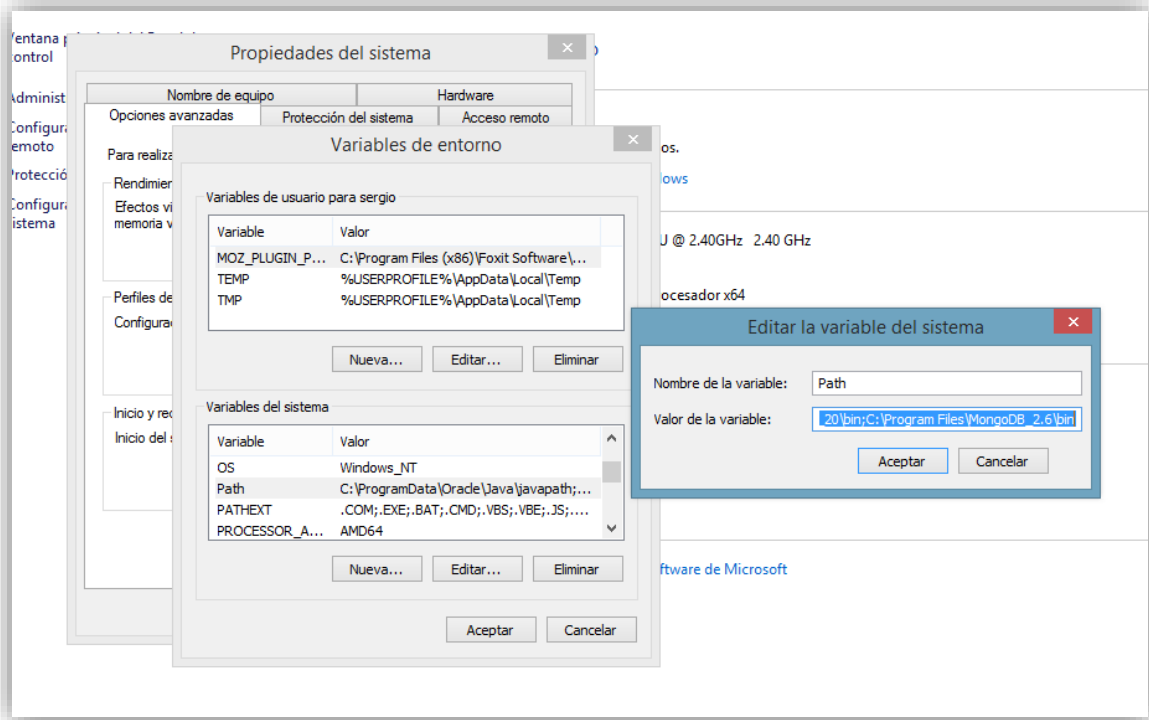
1-Descargar MongoDB desde aquí <http://www.mongodb.org/download> .

2-Para que la terminal reconozca la instrucción "mongod" es necesario añadir al path de Windows la ruta hasta la carpeta bin de mongoDB. En mi caso:

"C:\Program Files\MongoDB_2.6\bin"

Modificar el path de windows:

1. Panel de control -> Sistema y seguridad->Sistema -> Opciones avanzadas
2. Haga clic en Variables de entorno, en Variables del sistema, busque **PATH** y haga clic en él.
3. Añadir ";C:\Program Files\MongoDB_2.6\bin"



3- crear el directorio log y la carpeta data en "C:\Program Files\MongoDB_2.6" y el directorio db dentro de data

4- crear el archivo mongo.config con esto:

##store data here

dbpath=C:\Program Files\MongoDB_2.6\data\db

##all output go here

logpath=C:\Program Files\MongoDB_2.6\log\mongo.log

##log read and write operations

diaglog=3

Para iniciar la base de datos desde línea de comandos será del siguiente modo:

mongod -config mongo.config

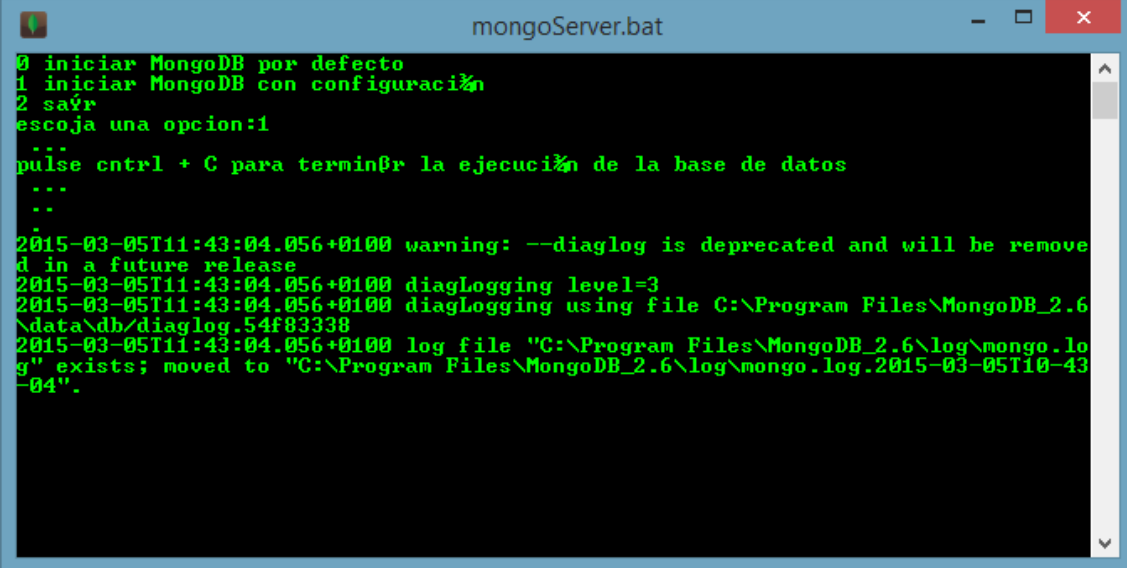
Estando en el directorio MongoDB_2.6 . o pasando la ruta absoluta en el caso de estar en otro directorio:

mongod -config /C:\Program Files\MongoDB_2.6\mongo.config

(Se proporciona un .bat que deberá estar en el directorio *MongoDB_2.6* para arrancar la base de datos.)

Es posible que debas ejecutar el bat como administrador dependiendo del directorio en el que instalases el MongoDB.

Una vez iniciada la base de datos deberás ver algo como esto:



```
mongoServer.bat
0 iniciar MongoDB por defecto
1 iniciar MongoDB con configuraci3n
2 salir
escoja una opcion:1
...
pulse cntrl + C para terminar la ejecuci3n de la base de datos
...
2015-03-05T11:43:04.056+0100 warning: --diaglog is deprecated and will be remove
d in a future release
2015-03-05T11:43:04.056+0100 diagLogging level=3
2015-03-05T11:43:04.056+0100 diagLogging using file C:\Program Files\MongoDB_2.6
\data\db/diaglog.54f83338
2015-03-05T11:43:04.056+0100 log file "C:\Program Files\MongoDB_2.6\log\mongo.lo
g" exists; moved to "C:\Program Files\MongoDB_2.6\log\mongo.log.2015-03-05T10-43
-04".
```

Tambi3n puede descargar la versi3n portable que se ejecuta simplemente haciendo doble clic sobre el ejecutable (.exe).

No obstante esta versi3n est3 m3s limitada y preconfigurada.