**Escuela de Ingeniería Informática de Oviedo**

2014/2015

**Trivial 4A**

**Documentación de la Arquitectura**

Arquitectura del Software

Contenido

[Planteamiento del problema 3](#_Toc412914174)

[Stakeholders 4](#_Toc412914175)

[Responsables de NoGame 4](#_Toc412914176)

[Desarrolladores de NoGame 4](#_Toc412914177)

[Usuarios de la aplicación 4](#_Toc412914178)

[Atributos de Calidad 5](#_Toc412914179)

[Disponibilidad 5](#_Toc412914180)

[Modificabilidad 5](#_Toc412914181)

[Seguridad 5](#_Toc412914182)

[Testabilidad 5](#_Toc412914183)

[Usabilidad 5](#_Toc412914184)

[Lista de Atributos de Calidad 6](#_Toc412914185)

[Escenarios de Calidad 7](#_Toc412914186)

# Planteamiento del problema

La empresa NoGame dedicada a la creación de videojuegos va a crear una nueva línea de productos basados en juegos de preguntas/respuesta para diversas plataformas. En primer lugar se desea construir una variante del juego Trivial, aunque la idea es que puedan crearse más juegos de preguntas/respuesta similares en el futuro.

En una primera fase, desean hacerse con una base de preguntas suficiente para poder alimentar los diferentes juegos. Para ello, desean construir una sencilla aplicación que lea ficheros con los enunciados y respuestas de las preguntas, procese dichas preguntas indicando si hay errores o son correctas y las almacene en una base de datos.

Los ficheros con los enunciados de las preguntas se tomarán de bancos de preguntas ya existentes. Uno de los bancos de preguntas utiliza el formato GIFT. La aplicación utilizará una representación interna de las preguntas en formato JSON que facilitará su posterior almacenamiento en la base de datos. La compañía está considerando utilizar MongoDb, aunque esta decisión todavía no se ha tomado.

Se desea que la aplicación pueda ejecutarse en dos etapas. Una primera etapa que analiza los ficheros con las preguntas y genera el formato intermedio y otra etapa que toma las preguntas en dicho formato intermedio y las almacena en la base de datos. La ejecución de estas 2 etapas será controlada por un operador de la compañía que podrá decidir cuándo ejecuta cada etapa e incluso podrá automatizar dicha ejecución para que se realice cada cierto tiempo.

Es importante garantizar que la conversión es correcta, facilitando la depuración del proceso de carga de preguntas. Por ese motivo, se buscará una solución que no requiera interfaz interactivo pero que permita varias opciones de entrada (elegir nombre del fichero a cargar, elegir formato de la entrada, elegir nombre del fichero de salida, elegir formato del fichero de salida, etc.) y que permita observar los resultados intermedios de la conversión para chequear que no hay errores.

Puesto que la conversión puede automatizarse para que se realice en cualquier momento, 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. Por ejemplo, la compañía está pensando procesar también preguntas en formatos XML como el formato QTI.

# Stakeholders

### Responsables de NoGame

Se trata de los directivos de empresa.

Son responsables de los presupuestos y toman las decisiones al respecto, buscando un bajo coste de desarrollo, es decir, ajustar el tiempo de desarrollo del proyecto y el coste del mismo.

### Desarrolladores de NoGame

Serán los responsables de desarrollar el sistema resultante de la arquitectura.

#### Desarrolladores del Parser

Desean construir una aplicación sencilla, que lea y procese los datos, y posteriormente los almacene. Asimismo, buscan baja intensidad tecnológica en el proyecto, para obtener un proceso de desarrollo controlado y predecible. No se requiere que la conversión sea interactiva.

### Usuarios de la aplicación

Son los que utilizarán el sistema. Querrán hacerlo de manera sencilla y eficiente.

# Atributos de Calidad

### Disponibilidad

Disponibilidad del sistema para procesar datos 24x7.

### Modificabilidad

Facilidad de cambio: nuevos ficheros fuente, nuevos formatos de representación de las preguntas, posibles cambios de base de datos.

Escalabilidad del sistema: incremento del número de usuarios y de las preguntas almacenadas.

### Seguridad

Seguridad de los datos almacenados: preguntas y respuestas, datos de los usuarios.

### Testabilidad

Facilidad para probar la fiabilidad de sistema, garantía de que las conversiones (Parser) se han realizado correctamente.

### Usabilidad

Facilidad de uso para los usuarios finales del juego, y para los encargados del mantenimiento del sistema.

# Lista de Atributos de Calidad

|  |  |  |
| --- | --- | --- |
| Código | Descripción | Tipo de Atributo |
| AT001 | Disponibilidad del sistema para procesar datos 24x7. | Disponibilidad |
| AT002 | Facilidad de cambio: nuevos ficheros fuente, nuevos formatos de representación de las preguntas, posibles cambios de base de datos. | Modificabilidad |
| AT003 | Escalabilidad del sistema, por posible incremento del número de usuarios y de las preguntas almacenadas. | Modificabilidad |
| AT004 | Seguridad de los datos almacenados: preguntas y respuestas, datos de los usuarios. | Seguridad |
| AT005 | Facilidad para probar la fiabilidad | Testabilidad |
| AT006 | Facilidad de uso para los usuarios finales del juego, y para los encargados del mantenimiento del sistema. | Usabilidad |

# Escenarios de Calidad

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Escenario nº | Fuente de estímulo | Estímulo | Entorno | Artefacto | Respuesta | Medición de la respuesta | Atributo de Calidad afectado |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |