Grado en Ingeniería Informática del software

Grupo: Trivial 1A

Documentación entregable 1

Arquitectura del Software



Contenido

[1. Planteamiento del Problema 2](#_Toc413057026)

[2. Requisitos Funcionales 2](#_Toc413057027)

[3. Requisitos No Funcionales 3](#_Toc413057028)

[4. Identificación de los Interesados (Stakeholders) 3](#_Toc413057029)

[5. Atributos de Calidad 4](#_Toc413057030)

[6. Atributos de calidad e Interesados 4](#_Toc413057031)

[7. Escenarios de Calidad 5](#_Toc413057032)

[9. Vistas 6](#_Toc413057033)

1. 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.

**Insertar el diagrama contextual aquí.**

1. Requisitos Funcionales
2. Se le debe poder comunicar a la aplicación por consola de donde tiene que leer la información.
3. Se le debe poder comunicar a la aplicación por consola donde ha de devolver el resultado.
4. 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)
5. La aplicación debe permitir observar los resultados intermedios de la conversión para detectar errores.
6. La aplicación debe poder leer los ficheros en el formato intermedio (JSON) y almacenarlos en una base de datos.
7. Estas dos operaciones podrán ser realizadas individualmente por un operario
8. Estas dos operaciones se deben poder automatizar, programando su ejecución cada cierto tiempo.
9. 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.
10. 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 | Skateholders | Intereses |
| ST-01 | Responsables de NoGame | 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 | 1. Baja complejidad en los algoritmos de aleatorización de las preguntas. |
| ST-03 | Responsables de la información | 1. Proyecto rentable, esto es, que permita ser desarrollado por el precio establecido con un grado de rentabilidad que haga atractivo el desarrollo. 2. Permitir la ampliación de la aplicación con nuevas partes de manera sencilla al proyecto. 3. Máxima facilidad y rapidez a la hora de introducir preguntas, preferiblemente con tratamientos por lotes y no individualmente. 4. Máxima facilidad y rapidez para buscar una determinada pregunta. 5. Máxima facilidad y rapidez para actualizar los datos de una pregunta. 6. Poder mantener la integridad y coherencia en la información. |

1. 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 los cálculos 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. | Estabilidad |
| 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 |

1. 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 |

1. Escenarios de Calidad

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

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Escenario | Fuente de estímulo | 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 el 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 |

9. Vistas