Requisitos no funcionales

Trivial5b - Versión 2

En la primera versión de la documentación extraíamos los requisitos no funcionales que albergaban mayor importancia en nuestra primera parte del sistema, un sistema de extracción y transformación de datos con arquitectura Bash. En esta segunda versión, añadiremos a los anteriores, los nuevos atributos de este nueva versión del sistema en la que además tendremos una aplicación de escritorio.

**Testabilidad**

Facilidad para probar y comprobar la fiabilidad del sistema, con garantía de que las preguntas han sido transformadas correctamente en ambas etapas del proceso, y que el juego funciona como se espera en todo momento.

**Integridad**

Los datos de la aplicación deben estar siempre consistentes, completos y correctos.

**Modularidad**

El sistema actual cuenta con dos módulos del sistema completamente distintos y los cuales son independientes el uno del otro, por lo que han de coexistir entre ellos.

**Integración**

Es un atributo importante de nuestro sistema ya que se pretende integrar el sistema de extracción y transformación de datos en un nuevo sistema más complejo que comprende una aplicación de escritorio. La integración de los datos abarca guardar los datos que una aplicación manipula de manera que otras aplicaciones puedan acceder a ellos.

**Modificabilidad**

El sistema ha de permitir la obtención de los datos de entrada con diferentes formatos por lo que se desea la posibilidad de incluir mayor funcionalidad en este aspecto en el futuro.

**Disponibilidad**

El operador encargado de la ejecución de las distintas etapas del proceso ha de ser capaz de iniciar cualquiera de ellas en cualquier momento que desee. En el caso del usuario, el juego debe de estar disponible para ser ejecutado en cualquier momento que este desee.

**Robustez**

Es la habilidad de un sistema para tratar los errores durante el proceso de ejecución. En nuestro caso sería, por ejemplo, el hecho de que en el proceso de tratamiento de los datos, tanto lectura como transformación, pudiera existir algún error y el usuario no debe ser consciente de que esto ha ocurrido, de manera que el sistema trate el error y continúe con una ejecución normal.

**Usabilidad**

En la anterior versión, el operador encargado de la ejecución de las distintas etapas cuente con una interfaz sencilla y usable. En la actual versión, además, siendo una aplicación de escritorio, cobra una mayor importancia este requisito ya que necesitamos que la interfaz del usuario sea sencilla e intuitiva.

**Portabilidad**

Se prevee que posteriormente se pueda crear una aplicación web similar a la actual aplicación de escritorio, por lo que pretendemos portar la aplicación a una diferente taxonomía.

**Accesibilidad**

Aparece una nuevo requisito extremadamente importante. La accesibilidad es la habilidad de un producto software de ser fácilmente accesible por una gran cantidad de aparatos tecnológicos y de distintos tipos de usuarios.