



Trabajo Práctico Especial

72.32 Diseño y Procesamiento de Documentos XML

Primer Cuatrimestre de 2024

Grupo 13

Integrantes:

Jonathan Lucas Blankleder - 64660

Ignacio Gastón Frund - 61465

Agustín Germán Ramírez Donoso - 64716

Introducción

Como la consigna sólo presentó ligeras modificaciones respecto a la del parcial, el presente informe también cuenta solo con algunas modificaciones respecto al entregado previamente.

Este trabajo consistió en la recuperación de información y estadísticas de los pilotos de la competición automovilística estadounidense NASCAR de cierto año y categoría, en formato XML desde una base de datos deportiva externa mediante una API ofrecida por la misma, para luego filtrar dicha información mediante consultas XQuery y generar un documento XSL FO que finalmente convertiremos en PDF. Junto al funcionamiento interno se debieron controlar los posibles errores y se creó un script bash que genera el archivo PDF automáticamente a partir del año y categoría solicitados.

Desarrollo del trabajo

Trabajo Realizado

Adjunto a este informe se encuentra el script bash *tpe.sh* que realiza todo el proceso de consulta y presentación automáticamente a partir de un *year* y *type* apropiados pasados por parámetro y mediante una API key declarada previamente. En caso de que no sean apropiados o la key no sea correcta, se imprimirá el error.

El mismo obtiene los archivos *drivers_list.xml* y *drivers_standings.xml* con la herramienta cURL a partir de dichos parámetros y la clave de la API de SportRadar creada específicamente para el trabajo. Además, a ambos archivos se les quita el Namespace para utilizarlos sin problemas.

Una vez que cuenta con ambos archivos, realiza la consulta *extract_nascar_data.xq*, que crea el archivo *nascar_data.xml* con la información que queremos mostrar y lo valida con el XSD proporcionado por la cátedra.

En base a este documento XML, el archivo XSLT *generate_fo.xsl* crea el archivo *nascar_page.fo* con el estilo solicitado, para luego exportar la información a un documento PDF mediante Apache FOP.

Es importante recalcar que en la entrega se incluyen el parser, el procesador y el generador de PDF ofrecidos por la cátedra (Xerces, Saxon y Apache FOP respectivamente) ya que el script utiliza un classpath local para ellos debido a que debimos correr el script con WSL y no podemos declarar las variables de entorno necesarias para que los mismos funcionen con nuestros archivos locales de dichos programas dentro de WSL. Sin embargo, esto trae el beneficio de poder correr el script sin instalarlos previamente, ya que sólo es necesario Java.

Dificultades

- Código en bash: si bien dentro de la carrera de Ingeniería Informática todos los integrantes del grupo vimos y fuimos evaluados en bash, el contenido limitado que vimos junto a la falta de práctica desde entonces provocó que el diseño del script fuera más complicado y tardara más de lo previsto. Además, debido a que todos somos usuarios de Windows, utilizamos WSL para ejecutar el script pero tuvimos problemas con las variables de entorno para el parser y procesador utilizados durante la materia. Por ello, se incluyeron los mismos ya que el script utiliza un classpath local.
- Atributos duplicados: durante la creación del archivo *nascar_data.xml* a partir de la consulta especificada, obtuvimos un error para ciertos parámetros donde un atributo de nombre “name” estaba duplicado. El problema era con los pilotos con más de un auto, pues estábamos seleccionando todos los nombres de fabricantes de todos sus autos. Lo solucionamos eligiendo sólo el nombre del primer auto que se procesa.
- Ausencia del valor en el campo rank: notamos que si bien los pilotos estaban ordenados según su puntaje de la temporada, su rango no figuraba en el documento final. El problema era una condición, donde utilizamos un if que no cumplía con lo que necesitábamos y lo reemplazamos con un choose.
- Condicionales en XQuery: la sintaxis de if-then-else en XQuery nos causó problemas a la hora de procesar los errores, ya que abordamos la situación como lo resolvemos en C o Java pero en este caso no funcionaba. La solución que encontramos no es elegante, pero funciona.

Aspectos investigados

- Programación en bash: como se mencionó en el punto anterior, debimos recurrir a notas y trabajos viejos para poder realizar el script necesario para automatizar los comandos de la consola y verificar la validez de los parámetros ingresados por el usuario. También debimos investigar sobre las variables de entorno, ya que los programas ofrecidos por la cátedra los necesitan pero WSL no los conoce aunque estén en el mismo sistema.
- Creación y uso de la clave API: debimos entrar en el apartado de desarrolladores de la página web de SportRadar para obtener esta clave, que luego debimos utilizar en partes específicas del trabajo para recuperar la información de los pilotos.
- Creación del PDF: para poder pasar la información de la consulta a un PDF, debimos investigar sobre XSL-FO para poder generar un archivo que luego, tras instalar Apache FOP, podemos convertir a dicho formato.

Roles

- Responsable del funcionamiento de la consulta XQuery: Jonathan Lucas Blankleder, Ignacio Gastón Frund.
- Responsable del funcionamiento de la plantilla XSLT: Jonathan Lucas Blankleder, Agustín Germán Ramírez Donoso.
- Responsable del funcionamiento global del proyecto: Ignacio Gastón Frund, Agustín Germán Ramírez Donoso.
- Responsable de la presentación: Ignacio Gastón Frund.

Conclusiones

En esta segunda entrega del trabajo dispusimos más tiempo al mismo, aunque aún así hicimos grandes partes a destiempo, es decir, por lo general uno trabajaba mientras los demás estudiaban para sus respectivos exámenes y luego, con git, traía las actualizaciones y continuaba con el trabajo. A pesar de eso, con el tiempo extra dedicado al mismo consideramos que esta segunda entrega se encuentra mucho mejor que la anterior.

De todas formas nos encontramos en descontento por el esfuerzo extra requerido para que el script funcione correctamente debiendo usar tantos programas externos donde las instrucciones de la cátedra son instalar variables de entorno que dependen del sistema operativo utilizado. Sí consideramos que los tutoriales ofrecidos son útiles, pero no cubrían la totalidad de los problemas que enfrentamos y por ello debimos investigar más allá de los requerimientos del trabajo y la materia, ya que la misma se trata de documentos XML, no de bash a un nivel más profundo que el visto en sus correlativas.