Universidades de Burgos, León y Valladolid

Máster universitario

Inteligencia de Negocio y Big Data en Entornos Seguros

Arquitectura *Big Data* de colas para el procesado de vídeo en tiempo real

Presentado por José Luis Garrido Labrador en Universidad de Burgos — 25 de febrero de 2020

Tutor: Dr. Álvar Arnaiz González y Dr. José Francisco Díez Pastor

Universidades de Burgos, León y Valladolid


Máster universitario en Inteligencia de Negocio y Big Data en Entornos Seguros

Dr. D. Álvar Arnaiz González, profesor del departamento de Ingeniería Informática.

Expone:

Que el alumno D. José Luis Garrido Labrador, con DNI dni, ha realizado el Trabajo final de Máster en Inteligencia de Negocio y Big Data en Entornos Seguros titulado Arquitectura *Big Data* de colas para el procesado de vídeo en tiempo real.

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 25 de febrero de 2020

Vº. Bº. del Tutor:	Vº. Bº. del co-tutor:
D. nombre tutor	D. nombre co-tutor

Resumen

En este primer apartado se hace una **breve** presentación del tema que se aborda en el proyecto.

Descriptores

Palabras separadas por comas que identifiquen el contenido del proyecto Ej: servidor web, buscador de vuelos, android ...

Abstract

A **brief** presentation of the topic addressed in the project.

Keywords

keywords separated by commas.

Índice general

Índice de figuras

Índice de tablas

Memoria

Introducción

Descripción del contenido del trabajo y del estrucutra de la memoria y del resto de materiales entregados.

Objetivos del proyecto

Los objetivos del proyecto se han dividido en tres apartados siendo estos los objetivos generales, los técnicos y los personales.

2.1. Objetivos generales

- Exploración de las diferentes herramientas para el procesado de vídeo en tiempo real a través de las fases de emisión, recogida, encolado, ingestión, procesado, enriquecimiento y almacenamiento.
- Estudio del estado del arte en análisis de imagen para problemas de salud ante distintos escenarios tanto en aspectos físicos (iluminación, enfoque...) como en aspectos lógicos (resolución, tasa de refresco...).
- Implementación del software necesario para la recogida de vídeo en tiempo real sobre sistemas de videoconferencia.

2.2. Objetivos técnicos

- Crear una infraestructura software basada en contenedores *Docker* para ser independientes del software anfitrión.
- Desplegar un *pipeline* sobre herramientas de la suite de *Apache* para el *Big Data* que satisfagan el flujo ETL propuesto (TODO Cita al flujo).
- Desarrollar algoritmo sobre *Spark Stream* que procese los vídeos generando los datos necesarios para los estudios posteriores.

2.3. Objetivos personales

- Contribuir a la mejora de la calidad de vida a través de facilitar soportes para la rehabilitación de pacientes de Parkinson.
- Conocer más profundamente las herramientas de la suite de *Apache* y como estas se pueden combinar para facilitar tareas de *Big Data*.
- Completas mi formación durante el máster a través de la creación de una solución que utiliza gran parte de los conocimientos adquiridos durante el mismo.

Conceptos teóricos

En aquellos proyectos que necesiten para su comprensión y desarrollo de unos conceptos teóricos de una determinada materia o de un determinado dominio de conocimiento, debe existir un apartado que sintetice dichos conceptos.

Algunos conceptos teóricos de LaTeX[1].

3.1. Secciones

Las secciones se incluyen con el comando section.

Subsecciones

Además de secciones tenemos subsecciones.

Subsubsecciones

Y subsecciones.

3.2. Referencias

Las referencias se incluyen en el texto usando cite [**?**]. Para citar webs, artículos o libros [**?**].

[1] Créditos a los proyectos de Álvaro López Cantero: Configurador de Presupuestos y Roberto Izquierdo Amo: PLQuiz

3.3. Imágenes

Se pueden incluir imágenes con los comandos standard de LaTeX, pero esta plantilla dispone de comandos propios como por ejemplo el siguiente:

Figura 3.1: Autómata para una expresión vacía

3.4. Listas de items

Existen tres posibilidades:

- primer item.
- segundo item.

1. primer item.
2. segundo item.

Primer item más información sobre el primer item.

Segundo item más información sobre el segundo item.

-

3.5. Tablas

Igualmente se pueden usar los comandos específicos de LaTeXo bien usar alguno de los comandos de la plantilla.

Herramientas	App AngularJS	API REST	BD	Memoria
HTML5	X			
CSS3	X			
BOOTSTRAP	X			
JavaScript	X			
AngularJS	X			
Bower	X			
PHP		X		
Karma + Jasmine	X			
Slim framework		X		
Idiorm		X		
Composer		X		
JSON	X	X		
PhpStorm	X	X		
MySQL			X	
PhpMyAdmin			X	
Git + BitBucket	X	X	X	X
MikTEX				X
TEXMaker				X
Astah				X
Balsamiq Mockups	X			
VersionOne	X	X	X	X

Tabla 3.1: Herramientas y tecnologías utilizadas en cada parte del proyecto

Técnicas y herramientas

4.1. Gestión de flujo

Uno de los puntos más esenciales de este trabajo es recoger y dirigir los *streams* de vídeo que se reciben. Por tanto, escoger una correcta aplicación para la gestión de este flujo de datos es esencial.

Dentro de la suite de *Apache* existen varios componentes que se encargan de la gestión del flujo de datos. Con el objetivo de que el sistema fuese lo más robusto, y siguiendo las recomendaciones del estado del arte (TODO citas de esto) se combinarían las herramientas siguiente.

- ***Apache Kafka*** [1] es un proyecto de intermediación de mensajes que trabaja sobre el patrón publicación-suscripción funcionando como un sistema de transacciones distribuidas. Incorpora para la implementación de este patrón un sistema de colas para la distribución de mensajes. Aporta una API para el productor, el consumidor, el flujo y el conector y la conexión se realiza a través del protocolo de la capa de transporte *TCP*.

- ***Apache Spark Streaming*** [2] es la extension sobre la API de *Spark* para la creación de aplicaciones sobre flujos de datos. Es un consumidor nativo de *Kafka*, *Flume*, los sistemas de ficheros *HDFS* y *S3* entre otras herramientas. El funcionamiento interno es a través de crear pequeños lotes de datos para pasarlo al motor de *Spark* y retornar los lotes procesados.

4.2. Infraestructura de bajo nivel

Otro apartado importante en el despliegue de la aplicación son las herramientas y técnicas a ser usadas para la producción. Para esto se utilizan:

- ***GNU/Linux***, el sistema operativo por excelencia en el entorno de los servidores [4, 5].
- ***Docker***, un software de gestión de contenedores estandarizados, semejante a los entornos *chroot* que facilita la virtualización de software en un entorno seguro y ligero. Sobre este motor se ejecutarán las aplicaciones del entorno de *Apache* [3]

Aspectos relevantes del desarrollo del proyecto

Este apartado pretende recoger los aspectos más interesantes del desarrollo del proyecto, comentados por los autores del mismo. Debe incluir desde la exposición del ciclo de vida utilizado, hasta los detalles de mayor relevancia de las fases de análisis, diseño e implementación. Se busca que no sea una mera operación de copiar y pegar diagramas y extractos del código fuente, sino que realmente se justifiquen los caminos de solución que se han tomado, especialmente aquellos que no sean triviales. Puede ser el lugar más adecuado para documentar los aspectos más interesantes del diseño y de la implementación, con un mayor hincapié en aspectos tales como el tipo de arquitectura elegido, los índices de las tablas de la base de datos, normalización y desnormalización, distribución en ficheros3, reglas de negocio dentro de las bases de datos (EDVHV GH GDWRV DFWLYDV), aspectos de desarrollo relacionados con el WWW... Este apartado, debe convertirse en el resumen de la experiencia práctica del proyecto, y por sí mismo justifica que la memoria se convierta en un documento útil, fuente de referencia para los autores, los tutores y futuros alumnos.

Trabajos relacionados

Este apartado sería parecido a un estado del arte de una tesis o tesina. En un trabajo final de máster no parece tan obligada su presencia, aunque se puede dejar a juicio del tutor el incluir un pequeño resumen comentado de los trabajos y proyectos ya realizados en el campo del proyecto en curso.

Conclusiones y Líneas de trabajo futuras

Todo proyecto debe incluir las conclusiones que se derivan de su desarrollo. Éstas pueden ser de diferente índole, dependiendo de la tipología del proyecto, pero normalmente van a estar presentes un conjunto de conclusiones relacionadas con los resultados del proyecto y un conjunto de conclusiones técnicas. Además, resulta muy útil realizar un informe crítico indicando cómo se puede mejorar el proyecto, o cómo se puede continuar trabajando en la línea del proyecto realizado.

Apéndices

Apéndice A

Plan de Proyecto Software

A.1. Introducción

A.2. Planificación temporal

A.3. Estudio de viabilidad

Viabilidad económica

Viabilidad legal

Apéndice B

Especificación de Requisitos

B.1. Introducción

B.2. Objetivos generales

B.3. Catalogo de requisitos

B.4. Especificación de requisitos

Apéndice C

Especificación de diseño

C.1. Introducción

C.2. Diseño de datos

C.3. Diseño procedimental

C.4. Diseño arquitectónico

Apéndice D

Documentación técnica de programación

D.1. Introducción

D.2. Estructura de directorios

D.3. Manual del programador

D.4. Compilación, instalación y ejecución del proyecto

D.5. Pruebas del sistema

Apéndice E

Documentación de usuario

E.1. Introducción

E.2. Requisitos de usuarios

E.3. Instalación

E.4. Manual del usuario

Bibliografía

[1] Apache Kafka. https://kafka.apache.org/.

[2] Spark Streaming - Spark 2.4.5 Documentation. https://spark.apache.org/docs/latest/streaming-programming-guide.html.

[3] Mario Juez-Gil. mjuez/spark-cluster-docker. https://github.com/mjuez/spark-cluster-docker.

[4] MuyLinux. Red Hat lidera el segmento Linux en el mercado de servidores. https://www.muylinux.com/2018/10/19/red-hat-lidera-mercado-linux-servidores/.

[5] Wensong Zhang et al. Linux virtual server for scalable network services.