Universidad de los Andes Colombia | Departamento de Ingeniería de Sistemas y Computación

Arquitecturas para Big Data

Curso de Verano	**CICLO LECTIVO: 2020**
NOMBRE: Arquitecturas para Big Data	**MODALIDAD: Online, teórico-práctico**
DURACIÓN: 4 semanas	**TOTAL DE HS: 36**

INSTRUCTOR: Dr. J. Andrés Díaz Pace

---

1. MOTIVACIÓN Y OBJETIVOS DEL CURSO

El desarrollo de sistemas que requieren procesar grandes volúmenes de datos, o integrarse con software de analítica o inteligencia artificial conlleva desafíos respecto a las técnicas y prácticas existentes de arquitectura de software. En particular, estos sistemas plantean nuevas perspectivas de atributos de calidad, patrones y tácticas de diseño, y arquitecturas de referencia, entre otros aspectos.

El curso introduce los conceptos principales del paradigma de Big Data desde una perspectiva de arquitecturas de software. El objetivo es proporcionar los fundamentos técnicos de diferentes tipos de técnicas, almacenamiento, y motores de procesamiento para Big Data. Adicionalmente, se discutirá un rango de mecanismos tecnológicos y modelos de referencia para la construcción de soluciones Big Data.

Como objetivos de aprendizaje, se pretende que el alumno logre:

- Comprender los conceptos principales, tecnologías, y patrones vinculados con Big Data en un contexto de desarrollo de software.
- Entender las implicancias de Big Data para el diseño de arquitecturas de software, la consideración de atributos de calidad, y su inserción en un proceso de desarrollo de software.
- Analizar los pros y contras de la incorporación de componentes y modelos Big Data en el diseño de una solución de software, y proponer distintas estrategias de adopción.

Los principales contenidos a cubrir incluyen: fundamentos de Big Data y consideraciones de adopción; relación con cloud computing; almacenamiento en disco y procesamiento batch; Map-Reduce; ecosistema Hadoop y principales mecanismos; tipos de almacenamiento NoSQL; almacenamiento en memoria y procesamiento en tiempo real; Bulk Synchronous Parallel; arquitectura Lambda; relación con sistemas empresariales, y esquemas de integración; relación con modelos de ciclo de vida.

CV breve del instructor: Jorge Andrés Díaz Pace es Doctor en Cs. de la Computación por la Universidad Nacional del Centro (UNCPBA, Argentina, 2004), y se desempeña actualmente como Profesor Asociado de la Facultad de Cs. Exactas de la UNCPBA, e Investigador Independiente de CONICET-Argentina. Trabaja en temáticas de Ingeniería de Software tales como: diseño de arquitecturas de software, asistentes inteligentes, Big Data y cloud computing. Desde 2007 a 2010 fue miembro del staff técnico del Software Engineering Institute (Carnegie Mellon University, Estados Unidos) en la división de arquitecturas de software. Posee certificaciones del Software Engineering Institute en análisis, diseño y evaluación de arquitecturas de software. Posee certificaciones en tópicos de cloud computing y Big Data otorgadas por Arcitura. Ha dictado cursos de posgrado en distintas universidades de Argentina y del exterior. Ha publicado varios trabajos de investigación en conferencias y revistas internacionales en temáticas de Ingeniería de Software. Ha brindado capacitaciones sobre estas temáticas tanto en Argentina como en el extranjero. Posee más de 12 años de experiencia como arquitecto consultor en proyectos de vinculación con distintas empresas de TI.

2. CONTENIDOS / UNIDADES TEMÁTICAS

Unidad 1: Arquitecturas de Software

Repaso de conceptos y terminología de arquitecturas de software. Objetivos de negocio y atributos de calidad. Escalabilidad. Disponibilidad y confiabilidad. Atributos de calidad para sistemas inteligentes. Patrones para almacenamiento y procesamiento de datos. Relación con cloud computing. Enfoque centrado en arquitectura (ACDM). Principales vistas de arquitectura y captura de decisiones de diseño.

Unidad 2: Fundamentos de Big Data

Definición de Big Data y dominios de aplicación en la industria. Datos generados por humanos y por máquinas. Principales motivadores tecnológicos y del negocio. Conceptos de dataset, analítica, inteligencia de negocios e indicadores clave de performance (KPIs). Las 5 Vs de Big Data. Tipos de datos estructurados y no estructurados. Metadatos. Tipos de análisis. Tipos de analítica. Consideraciones de adopción de Big Data en proyectos de software. Diseño de arquitecturas Big Data. Modelo de ciclo de vida integrado con Big Data.

Unidad 3: Almacenamiento y Procesamiento

Conceptos de sharding y replicación. Teorema de CAP. Principios ACID y BASE. Características del almacenamiento para Big Data. Sistemas de archivos distribuidos. Almacenamiento relacional. Tipos de almacenamiento NoSQL: documentos, clave-valor, tabular, y grafos. Procesamiento batch. MapReduce. Bases de datos en memoria y grillas de datos. Teorema SCV. Procesamiento real-time y near real-time. Modos de operación ESP y CEP. Spark y Storm. Bulk Synchronous Parallel (BSP). Diseño de pipelines de Big Data con procesamiento real-time y batch. Arquitectura Lambda. Patrones para Big Data.

Unidad 4: Sistemas Empresariales con Big Data

Conceptos de OLTP, OLAP, ETL y DataWarehouse. Arquitectura de un sistema empresarial tradicional. Posibilidades de mejora con Big Data. Tipos de arquitecturas de Big Data: de componentes, de solución, de integración, y empresarial. Esquemas de integración: serie, paralelo, producto integrado, e integración virtual. Arquitectura conceptual de referencia para Big Data. Gobierno de Datos.

3. BIBLIOGRAFÍA

- *Software Architecture in Practice*, 3rd Edition / Len Bass, Paul Clements, Rick Kazman / Addison-Wesley (2012).
- *Big Data Fundamentals: Concepts, Drivers & Techniques*. T Erl, W. Khattak, P. Buhler. Prentice Hall. 2015
- *Cloud Computing: Concepts, Technology and Architecture*. T. Erl. Prentice Hall. 2013
- *Streaming Data: Understanding the real-time pipeline.* A. Psaltis. 1st Edition. Manning Publications. 2017
- *NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence.* P. Sadalage, M. Fowler. Addison Wesley. 2012
- *Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems.* M. Kleppmann. O'Reilly Media. 2017
- *Materiales y apuntes complementarios aportados por el instructor.*

4. METODOLOGÍA

Los docentes explicarán de forma clara los contenidos del programa, procurando establecer los modelos y tecnologías básicas de Arquitecturas de Software y Big Data, que permitan a los estudiantes comprender dichos modelos para luego poder aplicarlos en situaciones concretas de proyectos. Para ello, se utilizarán distintos medios audiovisuales (por ej., powerpoint, video, pizarrón). El desarrollo de las clases se llevará a cabo a través de una exposición dialogada.

Se utilizará una modalidad de dictado virtual, utilizando las herramientas Zoom y Blackboard.

Los docentes proveerán la orientación bibliográfica a los alumnos, promoviendo el estudio independiente de las fuentes, sobre la base de los conceptos expuestos durante las clases.

Los alumnos deberán leer anticipadamente (en la medida de lo posible) los materiales, a fin de poder participar con aportes y consultas que surjan de esta actividad.

Dado que las clases serán de naturaleza teórico-práctica, se presentarán durante las mismas casos de estudio para su análisis y elaboración por parte de los alumnos. Estos casos de estudio apuntarán a la resolución de problemas y al desarrollo del pensamiento crítico.

5. DISTRIBUCION DE CLASES Y ACTIVIDADES PRÁCTICAS

CLASE	FECHA	HORAS SINCRÓ-NICAS	HORAS ASINCRÓ-NICAS	CONTENIDOS
1	Viernes 19/Junio 5:30-9:00pm	3	0	Unidad 1 - Contexto de Big Data - Posibles casos de estudio y ejercicios
2	Martes 23/Junio 5:30-9:00pm	3	3	Unidad 1 - Caso de estudio “Empresa de Transporte Público” (introducción) - Caso de estudio “Sistemas Bancario” (introducción)
3	Jueves 25/Junio 5:30-9:00pm	3	0	Unidad 2 - Caso de estudio “Empresa de Transporte Público” (continuación) - Ejercicio de ETL (por ej., Twitter a SQL)
4	Viernes 26/Junio 5:30-9:00pm	3	0	Unidad 2 - Caso de estudio “Sistemas Bancario” (continuación)
5	Martes 30/Junio 5:30-9:00pm	3	3	Unidad 3 - Continuación ejercicio de ETL (por ej., Twitter a MongoDB, o a Neo4J)

6	Jueves 2/Julio 5:30-9:00pm	3	0	Unidad 3 - Ejercicio con MapReduce (batch) - Ejercicio con Spark (opcional)
7	Viernes 3/Julio 5:30-9:00pm	3	0	Unidad 4 - Ejercicio de pipeline de procesamiento y ML (por ej., análisis de sentimientos en Twitter)
8	Martes 7/Julio 5:30-9:00pm	3	3	Unidad 4 - Ejercicio de procesamiento con ElasticSearch (opcional) - Ejercicio con Data Version Control (DVC)
9	Viernes 10/Julio 5:30-9:00pm	3	0	Evaluación

Total horas sincrónicas: 3 x 9hs = 27 hs.
Total horas asincrónicas: 9 hs. (aproximadamente 3hs. por semana)

6. CRITERIOS Y MODALIDAD DE EVALUACIÓN

El curso comprenderá evaluaciones parciales y una evaluación final.

Las evaluaciones parciales consistirán en el planteo semanal de preguntas referidas a temas centrales de cada unidad y a las correspondientes lecturas (por ej., pequeños cuestionarios en modalidad multiple-choice). Adicionalmente, los estudiantes deberán realizar ejercicios prácticos relacionados con problemáticas de Big Data y Arquitecturas de Software, y analizar casos de estudio presentados en clase.

Para las evaluaciones, se tendrán en cuenta los siguientes criterios:
- Claridad y precisión conceptual
- Establecimiento de relaciones pertinentes
- Coherencia entre la argumentación y el problema o cuestión planteada
- Aplicación adecuada de las técnicas

La participación en clase implica aportes informados que contribuyan a la discusión y al aprendizaje grupal.

Al final del curso, se realizará una evaluación final que abarcará los distintos contenidos desarrollados en las unidades temáticas, en función de los objetivos de aprendizaje planteados. Esta evaluación incluirá ejercicios de naturaleza teórica, pero también ejercicios de índole práctica.

La evaluación final será obligatoria e individual, pudiendo desarrollarse tanto en forma oral o escrita. Esta evaluación admitirá una instancia de recuperación.

La calificación final del curso (en una escala de 1 a 10) se obtendrá a partir de una combinación de las evaluaciones parciales, la participación en clase, y la evaluación final.