

Trabajo - Sistemas Distribuidos

(Primer Entregable - Iteración 1)

Este trabajo consta de 3 evaluaciones, que son:

- 1) Iteración 1, diseño del sistema distribuido y contestar preguntas respecto de las decisiones de diseño, valor 20%, fecha de entrega viernes 2023-04-07
- 2) Iteración 2, Implementación de funcionalidad, valor 40%, fecha de entrega viernes 2023-05-12
- 3) Iteración 3, Implementación de seguridad y/o protocolos de funcionamiento + presentación max 15 min, valor 40%, fecha de entrega miércoles 2023-06-21

La evaluación de este trabajo será iterativa, esto quiere decir que si en la “entrega 1” no entrego todo lo solicitado, esto debe estar presente en las siguientes entregas y así sucesivamente, deberá hacer las correcciones de ser necesario y/o podrá incorporar mejoras según estime conveniente

Este trabajo debe ser realizado de 5 o 6 personas por grupo, debe especificar a los miembros del grupo en el siguiente link

https://docs.google.com/spreadsheets/d/1_etsdnZKRL4Yr2tBNhISLjpdX4wHfkEQNcR9LONheC4/edit#gid=0

Iteración 1, Diseño de Sistema Distribuido

Entregable e indicaciones generales:

- Debe seleccionar uno de los problemas planteados más abajo y generar un documento pdf con el detalle del diseño, esto según especificación
- Debe crear un proyecto en github y dar permisos de colaborador al usuario luisveasc (luis.veasc@gmail.com). En este proyecto github consolidaremos todo lo referente al trabajo
- Dentro del proyecto debe generar una carpeta llamada documentación, en esta carpeta debe agregar el documento pdf con lo referente al diseño
- En fecha de entrega, un representante del grupo enviará un email a luis.veasc@inf.uach.cl, el asunto de este email debe decir: “sistemas distribuido - entregable X”, en el cuerpo del correo debe indicar la url del proyecto github y el grupo al que pertenece.
- La máxima hora de entrega para cada una de las iteraciones de este trabajo es a las 23:59hrs de la fecha señalada por iteración.

Todo diseño debe considerar:

- Debe presentar el diagrama de arquitectura de su solución, ejemplo: puede utilizar el diagrama UML de componentes de software o cualquier diagrama de arquitectura que muestre el diseño de su solución
- Listado cada componente de su solución con su respectiva descripción respecto de su proyecto, si utiliza herramientas (componentes de software) existentes, debe señalar la versión a utilizar.
- Debe proponer el modelo físico
- Debe proponer el modelo fundamental
- Debe considerar que el lenguaje de programación a utilizar es python, versión 3.6 hacia arriba
- Si considera necesario el desarrollo de una base de datos, debe presentar el modelo relacional (si es una base de datos relacional) o la lista de objetos de la base de datos nosql. También, debe presentar un diccionario de datos.
- Debe considerar que todo componente de software a desarrollar debe ser implementado con tecnología de container
- Debe considerar que en la tercera entrega (Iteración 3) se incorporará la comunicación codificada de los mensajes enviados entre componentes y/o protocolos que tengan directa relación con el funcionamiento del sistema implementado
- Realice un listado de preguntas que realizaría al cliente que está solicitando esta solución (como mínimo 10 preguntas), con las cuales podría complementar mejor su desarrollo. Establezca los supuestos que contesten dichas preguntas para que puedan realizar su diseño. Ejemplo: ¿qué sistema operativo utiliza?, resp: linux bajo la distribución Centos 8, ¿con cuantos servidores cuentas y cuales son sus características? contamos con 2 servidores con 16 core, 32GB RAM, 2TB de HDD.
- Considere que la organización no cuenta con infinitos recursos, por ende, espera aprovechar sus recursos de la mejor manera posible. Si presenta una solución que considere componentes y/o herramientas de pago (ejemplo: EC2 de AWS) debe presentar un presupuesto aproximado de los recursos a utilizar y los gastos que esto significa (mensualmente)
- ¿Qué tipo de herramientas utilizarán en su propuesta: software libre, de pago, desarrolladas por usted o híbridas?. Justifique su decisión ¿por qué tomaron esa decisión?, ¿qué ventajas y desventajas conlleva?
- Mencione bibliotecas de o frameworks de python que deberá utilizar para implementar su solución y de una pequeña justificación de por qué debe utilizarlo, ejemplo: utilizaremos la biblioteca Flask para implementar la capa de servicios REST.
- Si su solución contempla la incorporación de frontend, la implementación es libre, ejemplo: reactjs, vuejs, Angular, Flutter, etc.

Recomendaciones de herramientas de software libre: nginx, traefik, docker, kubernetes, redis, memcached, NATs, RabbitMQ, Lucene, Solr, Elasticsearch, mongodb, mariadb, postgresql, Couchbase, apache kafka, kubeMQ, Confluent

Problemas Propuestos

Problema 1) Sistema de archivos distribuidos que puedan ser utilizados como S3 de AWS, esto quiere decir, que los archivos almacenados deben ser accesibles como archivos expuestos en internet (ejemplo: un archivo en la carpeta public de un servidor web). Debe considerar un punto de carga el archivos (imágenes, pdf, docs, etc), luego de cargar de los archivos estos deben ser distribuidos a los distintos servidores destinos de su infraestructura. Además, debe existir una vista web centralizada que permita explorar todos los archivos almacenados y poder acceder a ellos mediante un link directo al archivo. Los archivos a manipular deben tener como máximo 5mb de tamaño.

Problema 2) Sistema distribuido para hacer backups de bases de datos, este sistema debe ser capaz de respaldar de forma centralizada bases de datos de diferentes servidores, en diferentes formatos, ejemplo: tienen 3 servidores con diferentes bases de datos, en server1 tiene db1(mariadb con la db de clientes de un supermercado), en server2 tiene db2(postgresql con la db de proveedores de productos alimentarios), server3 tiene db3 (mariadb con la db de un hospital), todas estas db deben ser respaldadas y deben ser almacenadas en un repositorio centralizado

Problema 3) Sistema distribuido que plantee una forma de solucionar el problema de venta de boletos de una plataforma como ticketmaster cuando existe una alta demanda de solicitudes. El sistema debe presentar una vista web en la cual el cliente pueda seleccionar el espectáculo al cual quiere acceder.

Problema 4) Capa de servicios WEB distribuida con autenticación, el objetivo de este sistema es disponibilizar una capa de servicios la cual puede ser consumida por diferentes clientes, los clientes serán otras aplicaciones que hacen uso de la información brindada por los servicios. La aplicación debe poseer un sistema de administración de dicha capa de servicios, por ende los administradores deben poder loguearse, brindar el acceso a los servicios a los clientes y deben poder ver estadísticas de uso de los servicios disponibilizados. Para realizar sus pruebas, luego de la implementación debería crear a lo menos dos aplicaciones clientes que hagan uso de los servicios disponibles.

Problema 5) Sistema distribuido basado en un buscador tipo google, este sistema debe ser capaz de cargar documentos de diferentes orígenes y entregar resultados de la búsqueda basados en índices invertidos, debe incorporar una memoria cache que ayude a agilizar las respuestas de las búsquedas. Concepto clave: web search engine.

Problema 6) Sistema de voluntariado con base de datos distribuida que permita a las personas ofrecerse como voluntarios en caso de un desastre de origen natural, esta base de datos debe distribuir la información basado en la edad de las personas, se espera que la db esté subdividida a lo menos en 3 grupos etarios, ejemplo: $X \leq 30$; $X > 30 \ \& \ X \leq 50$; $50 >$

X. Además el usuario debe agregar su ubicación geoespacial (es probable que en la segunda entrega se soliciten búsquedas respecto de este atributo). Se recomienda utilizar una base de datos distribuida como mongodb, mariadb o postgresql.

Problema 7) Sistema de mensajerías tipo whatsapp o telegram que permite múltiples usuarios, este sistema debe ser capaz de almacenar la información de las conversaciones en una base de datos distribuida, la cual almacene los datos de la conversación basado en el mes donde se realizó la conversación. El sistema debe poder consultar conversaciones pasadas, es decir, debe ser capaz de consultar la base de datos distribuida.

Problema 8) Sistema de suscripción a noticias de interés en sitios web específicos. El sistema debe ser capaz de almacenar múltiples usuarios. El sistema es capaz de notificar de alguna forma cuando se presenta una nueva noticia en los sitios de interés donde se suscribió un usuario, ejemplo: una notificación al celular, al email, mediante un bot de telegram o whatsapp, mediante una pwa. A modo de piloto debe crear al menos 3 sitios de noticias, donde al agregar una noticia de interés para un usuario, este reciba su notificación.

Problema 9) Sistema Streaming que se vaya procesando los datos a medida van llegando, debe contar con al menos 5 categorías de datos distintos, realice operaciones particulares de agrupación y consolidación de información, para luego presente gráficos estadísticas relacionados con estas operaciones, ejemplo: una tienda de retail tiene sucursales en todo chile (una sucursal en cada región), pero no todas las sucursales manejan el mismo sistema, este streaming busca consolidar información de ventas, sucursales, vendedores, etc, y basado en esta información mostrar resultados transversales de la tienda.

Problema 10) Sistema de archivos distribuidos de “gran” tamaño, los archivos a considerar debe pesar desde 20MB a 100MB, el objetivo de este sistema es particionar de forma homogénea respecto de los HDD disponible de cada servidor de su infraestructura, almacenar las partes del archivo según que corresponda, y luego debe permitir descargar o revisar el archivo completo, para esto deberá ser capaz de reconstruir el archivo y mostrar la lista de archivos disponibles en una vista web.

Problema 11) Proponga un sistema distribuido de su interés, la propuesta debe ser tratada a con el profesor más tardar el día jueves 23-03-2023 en horario de atención a estudiantes (9:50 - 11.20hrs)