

Prueba 1, Sistemas Distribuidos - Práctica 2024-01

Esta prueba consta de dos secciones, una teórica y una práctica. Para la parte teórica tienen 70min. Para la parte práctica tiene hasta el domingo 28-04-2024, la entrega es vía email (luis.veasc@inf.uach.cl), a lo más dos personas por grupo. La calificación está determinada por la siguiente fórmula:

Calificación = Teoría (74%) + problema 1 Práctica(12%) + problema 2 Práctica(14%)

El asunto del email debe decir: "Prueba1, Sistemas Distribuidos, Práctica"

El cuerpo del email debe decir: los nombres de los integrantes

Adjuntar el código en un archivo zip llamado:

Si realiza el trabajo de forma individual \Rightarrow nombre_apellido(1).zip

Si realiza el trabajo \Rightarrow nombre_apellido(1)_nombre_apellido(2).zip

Se descontará 1 punto por cada día de atraso en la entrega de la parte práctica.

Problemas

1) **Problema 1 (HTTP), biblioteca digital distribuida por tipo de documento, este sistema debe ser implementado en python**

Implemente mediante un esquema maestro-esclavo un buscador distribuido, el cual permite distribuir la carga de trabajo desde un nodo maestro a diferentes nodos esclavos los cuales deben implementar las búsquedas sobre una base de datos, la distribución de la base de datos debe ser realizada por tipo de documento digital presente en la biblioteca y por lo menos debe presentar 4 tipos de documentos, es decir, el esquema debe presentar por lo menos 4 esclavos. Las operaciones a implementar son:

- (a) Buscar por título del documento digital, en el siguiente ejemplo estoy buscando "ecuaciones" y "diferenciales". Aquí debe implementar una función tipo broadcast, es decir, el maestro recibe la consulta, realiza la consulta sobre todos los esclavos, luego recibe la respuesta desde los esclavos y entrega el resultado en un arreglo con los datos encontrados

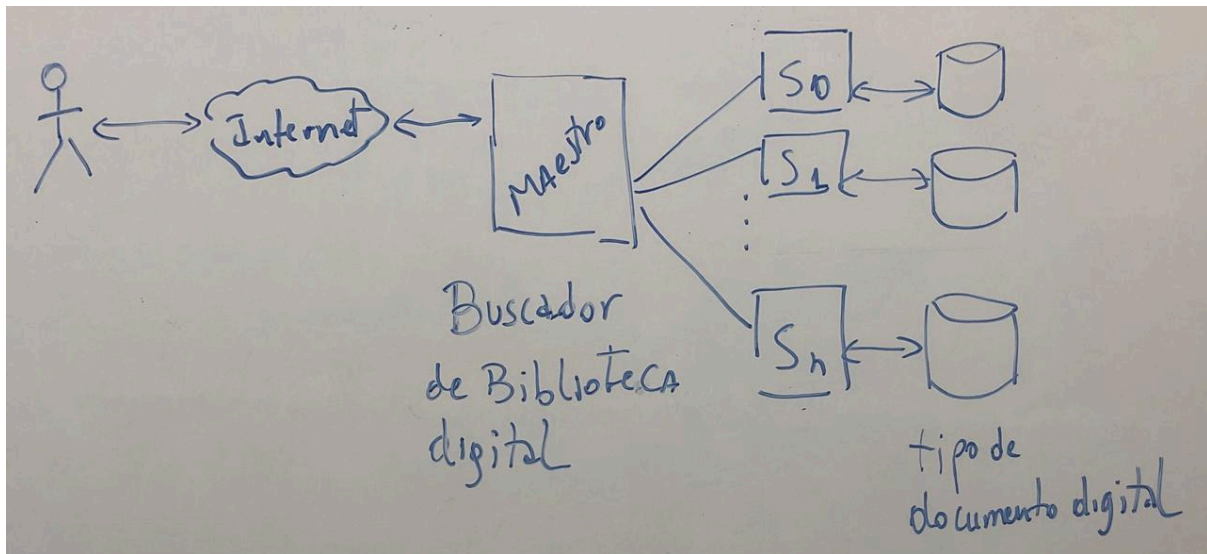
`http://localhost:xxx/query?titulo=ecuaciones+diferenciales`

- (b) búsqueda por tipo de documento, como el maestro conoce la distribución de la carga por tipo de documento, la consulta debe ir dirigida sólo a los esclavos que contienen la información, luego debe reunir las respuestas desde los esclavos y entregar el resultado en un arreglo con los datos encontrados

`http://localhost:xxx/query?tipo_doc=tesis+libro+video`

requisitos:

- El maestro conoce la distribución por tipo de documento alojadas en los sistemas esclavos.
- Esclavo tiene una copia de la base de datos con la porción de datos que le corresponda según la distribución por tipo de documento.
- Debe presentar un archivo README.md con la especificación de lenguaje, frameworks usados y la forma de despliegue. Si usa archivo de configuración o variables de entorno debe explicar su contenido.



2) Problema 2, juego basado en decisión por consenso del tipo p2p

El juego consiste en que dos o más equipos compiten por llegar a la meta en un tablero tipo fila de 100 posiciones, para lograrlo:

1. Cada integrante del equipo lanza un dado, el cual entrega una puntuación aleatoria de 1-6
2. Cada equipo tiene un turno de manera rotatoria

Requisitos:

- Debe presentar un archivo README.md con la especificación de lenguaje, frameworks usados y la forma de despliegue Si usa archivo de configuración o variables de entorno debe explicar su contenido.
- La cantidad de equipo máxima es variable, según sea definida por el juego
- La cantidad de integrantes máxima por equipo es variable, según sea definida por el juego
- La cantidad máxima de posiciones es variables, es decir, pueden ser más o menos de 100 posiciones
- La cantidad mínima y máxima del dado es variable, según se defina en el juego

- Gana el primer equipo que llega o supera 100 (o el máximo de posiciones, ya que es variable)
- La puntuación de avance depende de la suma de los puntajes obtenidos por cada miembro del equipo al arrojar el dado
- Los jugadores del equipo deciden si aceptan o no a otro integrante en el equipo, es decir, todos los integrantes del equipo se conocen

Condiciones de la implementación, debe implementar el sistema bajo un esquema cliente-servidor, donde:

- El servidor tiene 2 equipos por defecto y permite que se inscriban más equipos.
- El servidor inicia el juego después de que ambos equipos solicitan el inicio del juego, es decir, el juego es asíncrono
- El servidor de forma aleatoria asigna el orden de juego de los equipos.
- El servidor finaliza el juego y notifica al ganador a todos los integrantes de los equipos, esto lo hace cuando el equipo en su conjunto superó las 100 posiciones del tablero (o máxima cantidad de posiciones, ya que es variable)
- El servidor recibe la petición de un participante de unirse a un equipo en el juego, informa al equipo en cuestión que hay un interesado en unirse. Los miembros del equipo deciden si aceptan o no, esto puede pasar en cualquier instante del juego.
- El juego parte cuando hay por lo menos un participante por equipo.

