

Instituto Tecnológico de Costa Rica

Área Académica de Ingeniería en Computadores

Curso: CE-2103 Algoritmos y Estructuras de Datos II, GR 2

Proyecto #3 - MyInvincibleLibrary.

Realizado por:

Josué David Araya García - 2017103205

Luis Pablo Carmona Ulate -2018183396

Heizel Tatiana Chacón Mora - 2018203793

Luis Fabián Crawford Barquero - 2013052995

Profesor:

MGP. Luis Diego Noguera Mena

Fecha: Cartago, Mayo 29, 2019

Índice

Contenido

Análisis	3
Historias de usuario	3
Tareas por Historia de Usuario	4
Investigación	5
Propuesta principal	6
Propuesta secundaria	7
Trabajo en equipo	7
Roles de los integrantes.	7
Auto-evaluación y Co-evaluación	8

Análisis

Para la planificación de este proyecto se establecieron las bases mediante historias de usuario que sirvieron de guía durante el proceso de desarrollo y las cuales se detallarán a continuación.

Historias de usuario

- Historia de Usuario 1

Como usuario quiero un sistema de almacenamiento de archivos para guardar mis imágenes.

- Historia de Usuario 2

Como usuario quiero tener un sistema de seguridad para poder recuperar mis imágenes en caso de fallos.

- Historia de Usuario 3

Como usuario quiero recuperar mi imagen aún si la perdiera para ver en funcionamiento el sistema de seguridad.

- Historia de Usuario 4

Como usuario quiero tener un sistema de almacenamiento para la información de mis imágenes

- Historia de Usuario 5

Como usuario quiero gestionar la información mis imágenes para actualizar la información a mi gusto

- Historia de Usuario 6

Como usuario quiero gestionar mis imágenes desde una aplicación para facilitarme la interacción con mis programas.

Tareas por Historia de Usuario

- Historia de Usuario 1:

1. Crear los 4 discos virtuales, para implementar un RAID 5.
2. Crear una función de write.
3. Crear una función de read.
4. Crear una función de seek.

- Historia de Usuario 2:

1. Implementar un controlador para el RAID 5
2. Serializar la imagen a binario.
3. Dividirla en 3 porciones.
4. Crear una función para calcular la paridad de un archivo.
5. Guardar cada porción en un disco como jpg y en el disco sobrante guardar la paridad como un txt.

- Historia de Usuario 3:

1. Identificar si un disco fue eliminado.
2. Hacer un XOR que reconstruya la sección de la imagen del disco eliminado.
3. Hacer una función que muestre al usuario, solo cuando lo solicite, la imagen que el XOR reconstruyó.

- Historia de Usuario 4:

1. Implementar un base de datos no SQL, a través de una matriz 6xN.
2. En la matriz los datos se deben almacenar por filas, cada fila contiene todos los datos de una imagen.
3. Los datos a almacenar son: ID, nombre de la imagen, autor, año de creación, tamaño y descripción.

- Historia de Usuario 5:

1. Implementar una interfaz gráfica

2. Crear función agregar metadata.

3. Crear consultar metadata.

4. Eliminar metadata.

5. Modificar metadata.

- Historia de Usuario 6:

1. Hacer un IDE donde el usuario interactúe con la aplicación por medio de comandos.

2. Hacer un botón CARGAR IMAGEN para que el usuario pueda subir imágenes a la aplicación.

3. Mostrar la imagen seleccionada en la aplicación.

4. Hacer una tabla donde se muestren los datos que el usuario vaya a solicitar.

Investigación

Para la implementación de este proyecto se manejaron 2 opciones preliminares, las dos, aunque con mucha similitud, se distinguía una de la otra por el tipo de servidor a utilizar como medio de comunicación entre los componentes de My Invencible Library y el tipo de lenguaje.

Para ambas propuestas se planteó de la misma manera el desarrollo de un raid 5, una base de datos no relacional y un IDE.

El raid 5 se realizó de modo que tuviera capacidad de almacenar imágenes tipo png y no jpg, esto debido a que, de acuerdo a lo investigado, en diversos foros de programación se explica que en general el uso las imágenes en formato png es más recomendado en situaciones donde se tenga que manipular los bytes de la imagen, ya que si se utilizan imágenes tipo jpg esto podría provocar errores con la metadata.

Para My IDE se decidieron implementar los comandos como SELECT, DELETE o UPDATE que requirieran ingresar la menor cantidad de caracteres de manera que el proceso para obtener el texto en el IDE fuera necesario la menor cantidad de verificaciones y de esta manera tener la menor cantidad de comparaciones lo que a futuro nos puede ahorrar diferentes errores a la hora de correr el programa.

En la base de datos no relacional se utilizaron listas enlazadas, ya que estas nos permitían almacenar gran cantidad de información y por otro lado nos dan la posibilidad de manipularlas como matrices de modo que fue la manera más apropiada y funcional que se encontró para realizar una simulación de un base de datos.

La propuesta elegida involucra utilizar sockets como servidor y realizarlo junto con la lógica en el lenguaje de programación java. A diferencia de la segunda propuesta que se planteó que consistía en utilizar un Rest Api y que en general todo el proyecto se desarrollara en el lenguaje C++.

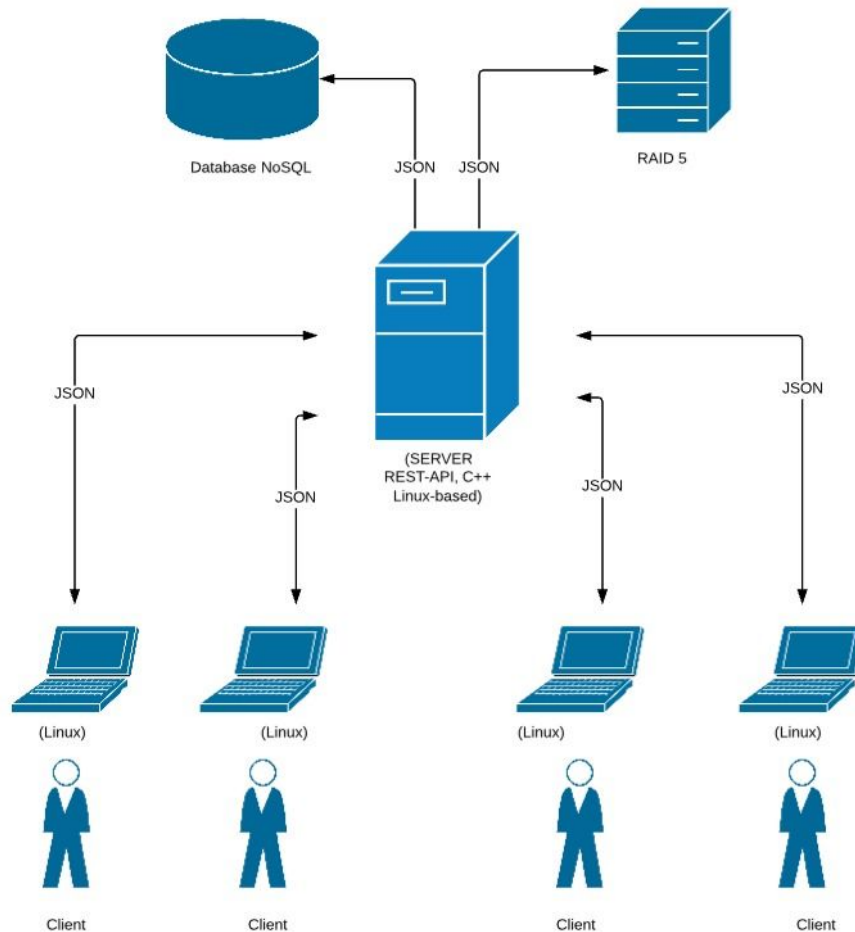
La decisión sobre el lenguaje se fundamentó principalmente en las facilidades que ofrece java, como por ejemplo todas la librerías disponibles para realizar las conversiones de imagen a bytes y viceversa, así como las opciones para guardar archivos en el ordenador.

Finalmente analizando la escogencia de los sockets sobre el Rest Api, la razón principal fue que debido el rest funciona por demanda, entonces se nos hace más sencillo que las solicitudes lleguen una a una y se manejen por medio de sockets. De esta forma se puede administrar a quien se le manda una solicitud y a quien no, por ejemplo, cuando el servidor recibe una solicitud y la redirige hacia el raid y la base. Solo basta con tener una clase que controle con quien se debe de conectar para mandar el mensaje y/o recibir una respuesta y luego seguir con el proceso.

Propuesta principal



Propuesta secundaria



Trabajo en equipo

Roles de los integrantes.

El equipo de trabajo se organizó a partir de las historias de usuario descritas anteriormente, de manera que se asignó ciertas tareas a cada integrante. Esto con la finalidad de poder tener una mejor administración y control del avance del proyecto.

Como líder del equipo de designó a Pablo Carmona, quien se encargó de el desarrollo del raid 5, por otra parte los integrantes Fabian Crawford y Tatiana Chacón se hicieron cargo de la implementación de Server Library y My IDE, finalmente Josué Araya se le asignó la tarea de realizar base de datos no relacional.

El equipo como parte de una correcta planificación acordó reunirse dos veces a la semana, tanto para poder verificar el avance de los compañeros, así como solventar las dudas y verificar que en general el proyecto llevara el rumbo adecuado.

Auto-evaluación y Co-evaluación

A continuación, se muestra la tabla que contiene el resultado de la auto-evaluación y co-evaluación de cada integrante del equipo, donde el rango va desde 0 como mínimo, hasta 10 como calificación máxima.

Integrante	Auto-evaluación	Co-evaluación
Josué Araya García	10	10
Pablo Carmona Ulate	10	10
Tatiana Chacón Mora	10	10
Fabian Crawford Barquero	10	10