**Algoritmos y Estructura de Datos –CC 182** 

**Título del proyecto:**

**Blockchain**

**Sección:**

CC31

**Integrantes:**

Eduardo Roman Caballero Lara (U202019644)

Franco Marcelo Manuel Surco Reyes(U202015132)

**Profesor:**

Heider Sanchez

**Ciclo:**

2022 - 00

**ÍNDICE**

* **Introducción**
* **Descripción del caso de estudio planteado por el grupo**
* **Diagrama de clases de entidades principales**
* **Definición de TDA y estructuras de datos a usar o diseño de archivos a utilizar**
* **Elaborar plan de trabajo detallado (Proyecto, Milestones, Issues)**
* **Complejidad en notación detallada y Big O de los métodos de las estructuras de datos.**
* **Conclusiones**
* **Referencias**

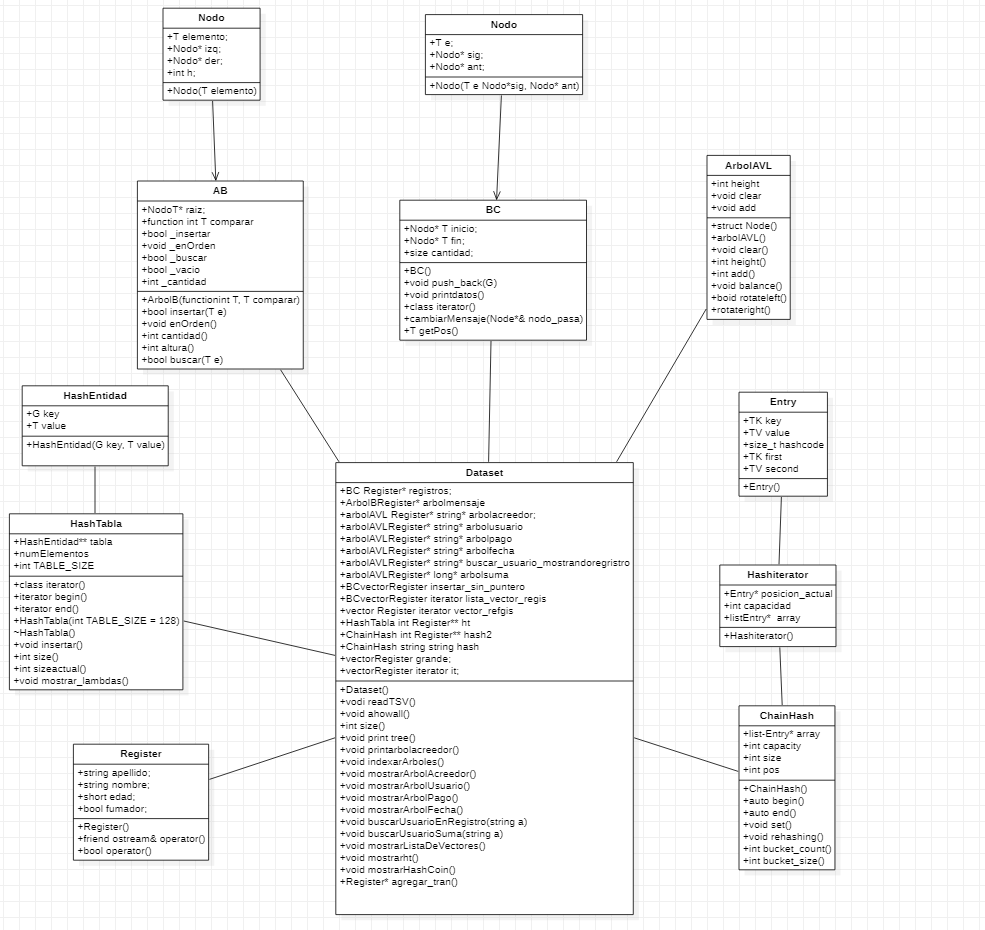
**Introducción**

El presente trabajo consiste en la implementación de una aplicación sobre una estructura de datos de cadena de bloques el cual contiene un único host. Tiene como funciones principales la búsqueda por ”keys” gracias a la función hash table, la carga y guardado de archivos en disco duro y funciones que permiten a los usuarios navegar entre sus propios datos. Este proyecto es gestionado en la plataforma de github y trabajado en visual studio además de utilizar distintas aplicaciones para editar nuestro código, este ha sido creado con uso de POO, estructuras de datos, lambdas y templates en una interfaz CLI.

**Descripción del caso de estudio planteado por el grupo**

Se nos pidió implementar, mediante el uso de listas dobles, hash table, templates y lambdas, una estructura de datos que sigue la filosofía de los árboles binarios. Esto con la finalidad de aminorar los tiempos computacionales al realizar las búsquedas. Además, cuenta con opciones de inicio de sesión para que cada usuario pueda cargar y crear archivos y solo ellos puedan verlos.

**Diagrama de clases de entidades principales**

****

**Definición de TDA y estructuras de datos a usar o diseño de archivos a utilizar**

Primero un concepto de TDA y una estructura de datos, para entender que son

TDA:Una representación que ha sido extraída, cuyos datos sólo puede ser accedida a un conjunto de operaciones.

Estructura de datos: Es la codificación(implementación ) de los TDA

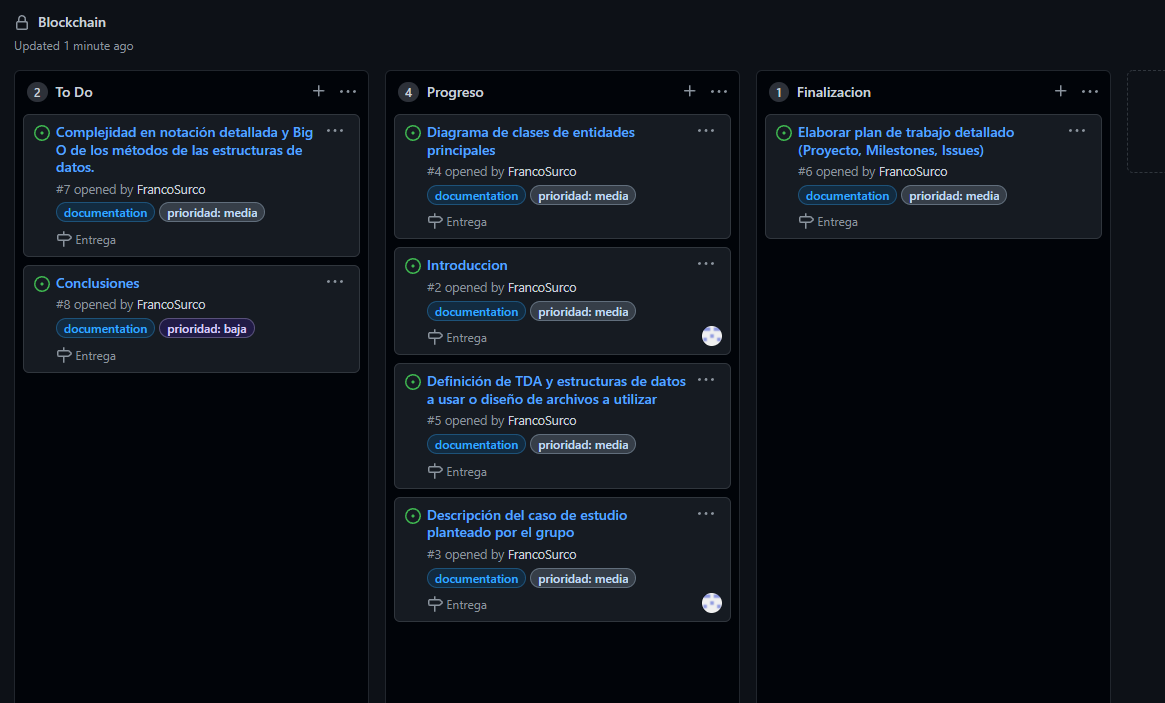
Sabemos que para el diseño del ciclo de vida del software, en este caso implementarlo en el proyecto, primero requiere una serie de pasos previos, primero por el análisis, luego el diseño y finalmente codificación. En este caso el TDA se encuentra en la parte del diseño y la Estructura de datos en la parte de codificación del programa a implementar.

Para nuestro proyecto utilizaremos TDA de tipo estructural, que será de tipo lineal para ser más precisos nos referimos a las listas dobles,de tipo directo hashtable y el otro de tipo log para indexar son los árboles binarios.Por lo tanto, sabemos que vamos extraer una parte del diseño para luego implementarlo a nuestra manera en este caso nuestra Estructura de Dato sera una version modificada de estas TDA mencionadas, para facilitar la codificación y la implementación de nuestros objetivos del blockchain.Con el uso de requeridas funcionalidades de la Listas,hashtable y los árboles .Además de otras funcionalidades empleadas en clase de las TDA para ser empleadas en este proyecto.

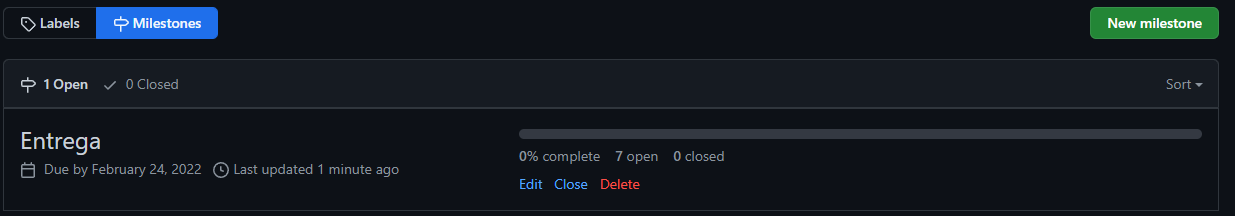
En los siguientes puntos mostraremos las estructuras mencionadas y sus tipos de datos.

**Elaborar plan de trabajo detallado (Proyecto, Milestones, Issues)**

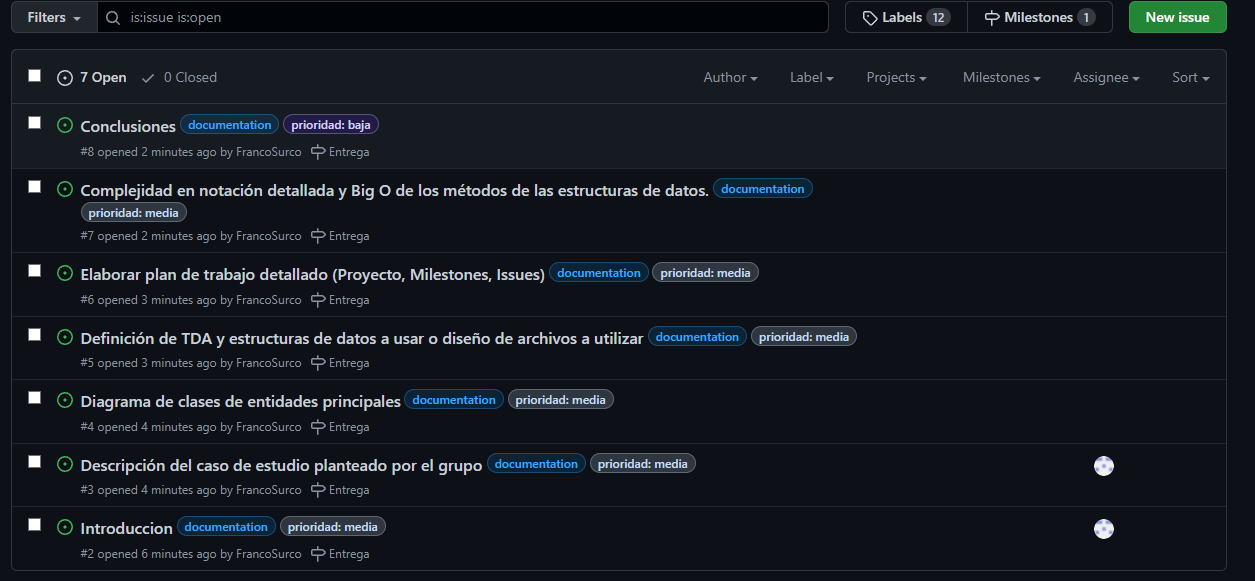
Se creó el proyecto BlockChain



Milestone



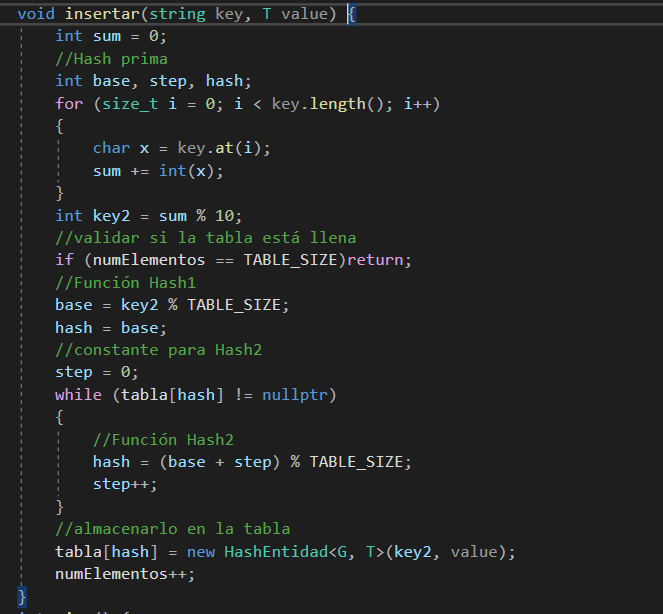
Issues



**Complejidad en notación detallada y Big O de los métodos de las estructuras de datos.**

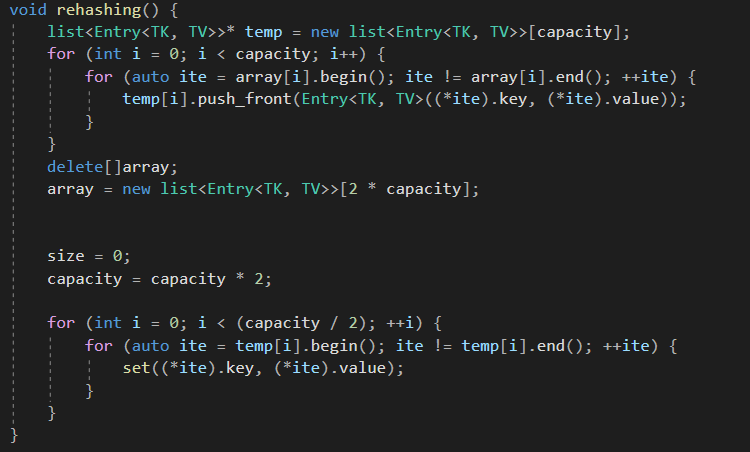
**HashTabla**

**Las inserciones son de tiempo 1**

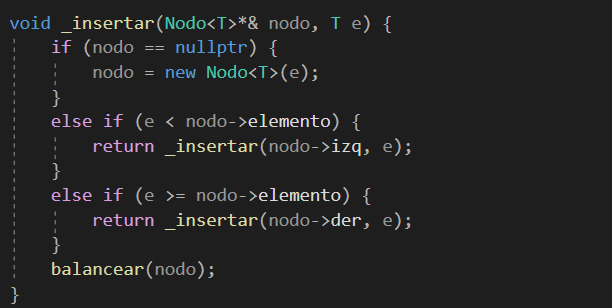
****

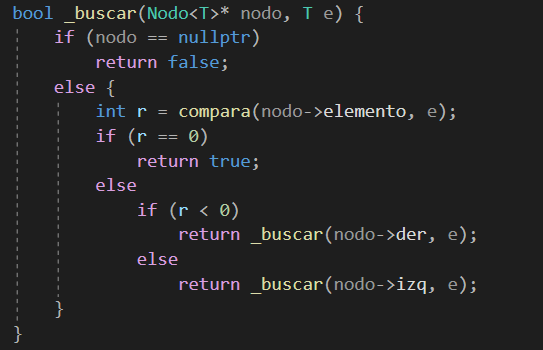
**ChainHash**

**Las rehash son de tiempo 2N**

****

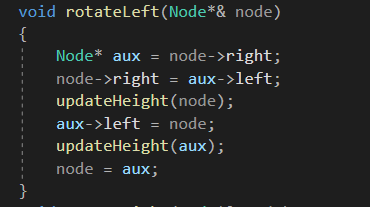
**ArbolB**

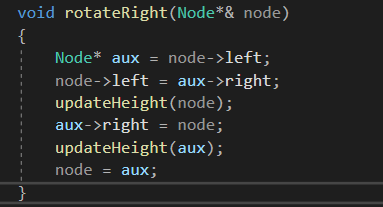
* **Las inserciones son de tiempo logN **
* **Las búsquedas son de tiempo logN**

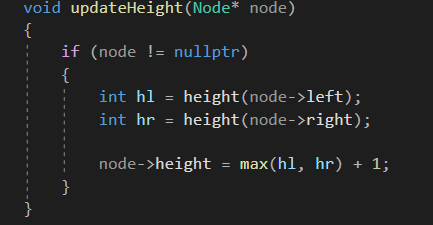
****

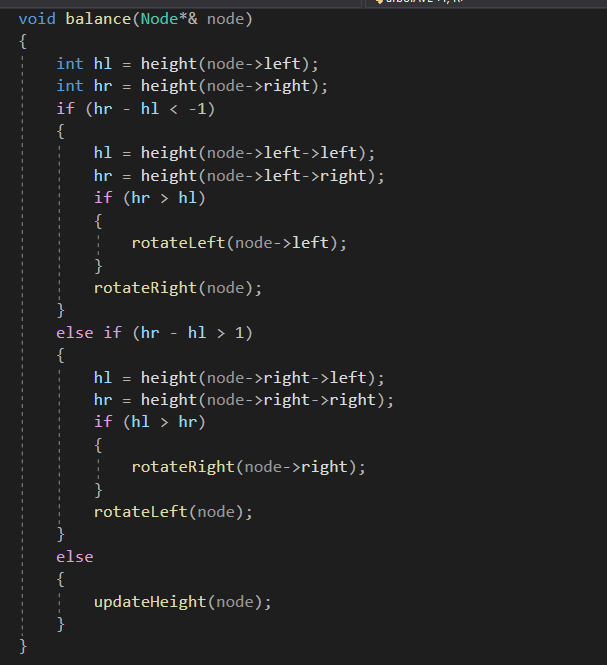
**ArbolAVL**

* **Las rotaciones y actualización de altura son de tiempo 1**

****

****

****

****

**Conclusiones**

Aplicamos los conocimientos aprendidos en clase para la resolución del proyecto. Para esto, hemos considerado la importancia de las TDA y estructura de datos para la implementación de nuestro proyecto. Además de aprovechar las ventajas de sus funcionalidades de los árboles y listas, ambas mencionadas en los anteriores puntos del presente trabajo. Adicionalmente, en el trabajo, hemos aprendido la función de indexar un árbol a una lista.

**Referencias**

-Walter Cueva Chavez. (2020, 5 junio). Gestión de Proyectos en Github [Vídeo]. YouTube. <https://www.youtube.com/watch?v=Vjf_s7TGmqY>

-Cormen, T. H., & Cormen, T. H. (2001). *Introduction to algorithms*. Cambridge, Mass: MIT Press.

Recuperado de: <https://courses.csail.mit.edu/6.006/spring11/rec/rec04.pdf>