Project Report

IPFS

December 14, 2023

Project Report for Interplanetary Filing system created by students from the third semester of FAST Nuces – Islamabad Campus

By

Muhammad Ali Naveed

Hamza Omer

Ariyan Chaudhary

Interplanetary Filing System (IPFS)

|  |
| --- |
| **Table Of Content** |

|  |  |
| --- | --- |
| **Content** | **Page No.** |
| 1. **Objective Analysis And Purpose** | **3** |
| 1. **Thought Process And Approach** | **4** |
| 1. **Explanation of Classes** | **5** |
| 1. **Conclusion And Final Results** | **5** |

Overview

The Interplanetary Filing System is a project assigned to third semester students studying the Data Structures and Algorithms course. This group is comprised of three students; Ali Naveed, Ariyan Chaudhary and Hamza Omer. They have devised an efficient and attractive solution to the problem statement utilizing the skills and techniques acquired throughout the semester. This project required some out of the box thinking, innovative solutions, a structured approach and most importantly, the time and dedication the project deserved. The project was conducted in a unified manner with all members of the team working together to complete each aspect of the project, making sure all operations are fully functional. However, each member ended up taking charge of specific aspects to ensure a fair distribution of the workload. The functionalities of the B-Tree were taken up by Mr. Ali Naveed, Concocting the very essential utility functions were Mr. Ariyan Chaudhary’s forte and Mr. Hamza Omer took upon the responsibility of managing the file handling tasks. The three then worked together to combine each specialty and compile it into one complete product.

Objective Analysis

The main objective of the project was to design an interplanetary filing system. This essentially means that the team had to implement a filing system so that a client can store their data across multiple machines geolocated anywhere in the world. Now, since the core concepts of Computer Networking, Routing and managing clusters of physical machine nodes was not in the pay-grade for these students, this project was to be implemented on a smaller scale, imitating machines as nodes in a linked list with folders depicting each machine. Although this was a scaled down version of the actual filing system, it would still provide the client with all necessary functionalities like adding machines to the system, adding files to the system and sending queries to different machines. The theory of routing tables was also imitated using linked lists and all operations were to be performed in O(log(n)) complexity. Completing tasks linearly was unacceptable as if would render the routing tables useless. The project simulation included a circular linked list, with its nodes depicting an individual machine. Each machine would have a routing table to be able to access other machines in the system, a B-Tree to store all the data of the files stored in each machine, and it would also contain the data of the machine itself, namely the machine name and machine id.

Purpose

The purpose of the project was to introduce students to real-world problems and give them a sneak peek into how these problems should be approached. It was an opportunity for the students to hone their programming skills and apply their minds to form innovative solutions and come up with newer and more effective ideas. Students at this level should be comfortable with linked lists, file handling, basic string manipulation and dealing with different number systems in real-world applications. This project incorporated all these fundamental and semi-advanced concepts and demanded a complete demonstration of the aforementioned skills in a very precise and structured manner.

Thought Process

As with any deliverable of this magnitude, a crystal-clear thought process was essential before starting any actual implementation. Extensively detailed discussions, long brainstorming sessions and a lot of planning resulted in an initial structure and approach to the project. This included identifying troublesome aspects of the project, the classes needed and the functionalities encompassed by each class. Furthermore, this stage of the project required a lot of research on different aspects such as filesystems, hashing algorithms and routing tables as these were essential concepts in the project and not exclusively taught in the course. Admittedly, ChatGPT proved to be a crucial asset at the research stage, helping out with basic functionalities such as creating directories, providing insights on the SHA-1 hashing algorithm including the encryption and decryption process, and many other minor inquiries.

Approach

Following up on the thought process, the team started with the basic structure of their code with help from their corresponding class diagram. A lot of the initial work relied on the class diagram and then they made more developments on top of that and kept updating their code and diagram side by side. The class diagram can be viewed [here](https://lucid.app/lucidchart/43e3198c-b69f-461a-94a7-b600ce4a60cf/edit?viewport_loc=-1275%2C-191%2C4542%2C2051%2CHWEp-vi-RSFO&invitationId=inv_dc1649f2-148d-4eae-be01-7da2b1510b06). As shown in the diagram, the identification of the classes and member functions was a crucial step in beginning work on the project. Naturally, as the code progressed, more issues were identified and dealt with and more functions were required. The team also ensured that the integrity of the code was maintained, with each class containing private member variables and incorporated concepts of aggregation and composition. Another key aspect of the code was that the user is not given access to any class, rather just the terminal-based interface with all functionalities integrated and all input validation checks implemented. In the latter stages of the project, Mr. Ali Naveed discovered a very innovative way to display our linked lists and trees, utilizing a tool named GraphViz. The relevant information about the tool is clearly mentioned in the documentation.

Explanation of Classes

The documentation for the classes and their member functions defines what each of them do and why they were important, however the explanation and rationale behind those specific classes is defined here. The main classes in the code were, IPFS, DHT, MachineNode, RFT and BTree. The IPFS class contains the basic user-friendly interface for performing all the necessary functions. The DHT class contains the basic ring DHT functions which includes the management of the circular linked list. This DHT contains a list of MachineNodes. Each MachineNode has its own B-Tree and Routing table which are generated and reconstructed after each machine addition and removal. The B-Tree is structured to accept any order and stores the paths and hashes of the files being stored in the machine. IPFS integrates the DHT class which in turn integrates the rest of the classes within itself. More information about each class can be found in the documentation attached.

Conclusion

This project has been incredibly helpful in the development of coding skills along with team building skills and working with other members in mutual understanding, respect and cooperation. It has been an opportunity to hone data structures skills in a structured manner and seeing a product come to life is a very rewarding feeling for any programmer. It provides the team with a sense of accomplishment and boosts morale.