Symform Hiring

Take home problems.

"Big Data" Problem

Design a backend store for storing metadata about files and directories belonging to different users as part of an online storage service. The logical data model is as follows:

- a user has zero or more files or directories
- a directory contains zero or more files or sub-directories
- a file or directory has a relative path that is up to 32KB in length
- a file has a size and last modified time

This store should handle the following online queries efficiently:

- lookups on single files or sub-directory given a user and relative path
- listing all files and sub-directories belonging to a user or at a given level of the hierarchy
- inserting new files and directories

It should also support the following batch operations that run on a daily basis:

- count the total number of files per user, and the aggregate file size.
- count the number of files and sub-directories added per day for each user.

The store should scale to Billions of files and directories, and Millions of users. Please submit a sketch design for this store, including choice of technology, schema, and an analysis of how the design achieves the stated goals.

Network Problem

Design and implement a program that transfers fragments of a file to multiple remote nodes. The class should take as input:

1. A file path to file of a fixed size (64MB).
2. An unordered list of 64 IP addresses/port numbers.
3. The maximum transfer rate for the sender (in Kbps). The rate limit describes the maximum aggregate bandwidth a sender consumes when transferring one or more fragments of the file.

The program should divide the file up into 64x1MB fragments and send one fragment to each of the 64 nodes. A TCP socket should be used for communication with each of the nodes. The transfer should honor the rate set by the user. Assume that each of the remote nodes have heterogeneous bandwidth. The focus should be on:

- Minimizing the overall time it takes to send the entire 64MB file.
- Minimizing the communication overheads.
- Minimizing system resources consumed during the transfer.

Please submit the solution in a modern OO language of your choice (but preferably C#), and include a one page (or less) description of why you chose the solution and an evaluation of how it performs against the goals stated above.

Web Design Problem

Design and implement a simple web application to do the following:

- Shows a file explorer to navigate set of root level folders.
- User should be able to search for specific files and subfolders.
- User should be able to create new root level or subfolders.
- User should be able to add/remove files.

Please submit the solution in a modern OO language of your choice (but preferably C#), and include a one page (or less) description of why you chose the solution and an evaluation of how it performs against the goals stated above.

Discus design tradeoffs around efficiency and user experience.

File Sync Problem

Design & Implement a simple file sync that will sync files/folders between any two local folders. It should handle files changing on any side as well as if they changed simultaneously on both sides.

Discuss what kinds of issues you need to handle to do this very reliably.

Now imagine if you had to sync between multiple folders and if those folders were across the Internet.   What additional challenges are posed and what design approach you would take and the associated set of tradeoffs you'll make.