Magic Leap Platform Assignment

Overview

This assignment is meant to demonstrate your understanding of what it takes to build modern, large-scale web applications. The brief below has some specific requirements and some things that are up to you. Keep track of your decisions and be able to justify them. Your development process is just as important as the final output.

Instructions

Imagine you've been tasked with building an app store for a new consumer electronics product. Build out the most basic prototype of this:

- one HTTP endpoint that accepts uploads and stores them in the cloud
- one HTTP endpoint that allows retrieval of an uploaded asset.

Deploy this system to the cloud provider of your choice, and provide us a link where we can test it out.

Requirements

- Provide an HTTP endpoint to upload assets and retrieve a unique identifier for the uploaded file.
- Provide an HTTP endpoint to download an asset by its identifier. The original filename should be preserved when downloading.
- Plus, pick one of the following:
  - Add user-based access control to your files such that only the user that originally uploaded the file can access it.
  - Add token-based access control to your files such that instead of the identifier, files can be accessed with a token that expires after a set period of time.
  - Add an endpoint that returns a list of all files in the system, their identifier, original filename, and the byte size of the file.

    - Build a web page/app that provides a browser-based mechanism for using your upload and download endpoints.
    - Automate the setup of all infrastructure (servers, cloud services, code, etc) such that you could easily deploy a second complete, working copy of your app in a command or two.
    - Write an automated test suite that proves your system works as expected.
- Deploy this code to the cloud provider of your choice and provide us a link (it can be password protected or otherwise authenticated if you'd like).
- Provide a link to your code from a git-based source control platform or bundle your code together in a ZIP so we can review it.
- Include a README that details
    - which additional requirement you chose
    - how to compile/build/run the code locally
    - where to access the deployed version of the project
    - all design / architectural / technical decisions

Further Information

- We recognize that this project is a time commitment on your part, outside of your regular work and personal obligations. There is no strict time limit, but we suggest you budget four to six hours to ensure you can build, deploy, test, and document your project. We heavily use AWS, Go, and JavaScript for our backend development. If you're familiar with any of these, we suggest using them. If not, that's okay too — it's more important that you produce a quality project using tools you are familiar with.
- If you have any questions, don't hesitate to reach out to your recruiter for guidance.