Scenario:

Current system:

We have a product that allows users to store documents containing data related to the patients participating in a clinical trial. This data takes the form of tabular database data (a nosql collection called “Documents”), text files and images (stored in s3). This product is composed of an API for handling file upload and viewing and crud for the metadata associated to the files. There is also a front end browser client to consumer this API.

New feature:

A customer requires an automated reporting feature. The reports require a long running process to aggregate the data from a database and a file store and do complex image processing. The resulting report data will be compiled into a pdf summary and emailed to the subscribed users at a regular interval. We have a very large user base and expect this new feature to be used heavily.

User stories:

- As a user I expect to subscribe to a weekly report
- As a user I expect the report to be sent via email

Technical Details:

Documents collection

```
{
      _id: ObjectId, // Unique identifier of the document db record
      Filename: String, // Human readable file name
      FileLocation: String, // Points to s3 Bucket and path
      Size: Number, // The size of the file in bytes
      MD5: String // The md5 hash of the file
}
```

API

POST /api/documents

- Creates a document with a file name and the binary data

GET /api/documents/{id}

- Gets the document db record

PATCH /api/documents/{id}

- Updates the document db record (any property except _id)

DELETE /api/documents/{id}

- Deletes the document db record and s3 file

GET /api/documents/{id}/file

- Download the document file

Task:

Design a system for the automated report feature

Expectations:

- Provide any diagrams you think appropriate define the system
- Define or amend the APIs involved in your solution
- Provide a written summary of the solution
- Be prepared to walk us through and explain the approach you took

We will analyze:

- System design
- Adherence to best practices

Notes:

- Document any assumptions you make
- Feel free to reach out if you have questions
- Please keep track of the time it takes for you to complete the exercise

Submittal:

When completed, send us a link to the GitHub repo with the documents and diagrams defining the system. In your message please give us the approximate time it took to complete the challenge

Questions:

If you have any questions you can contact Rastko at rastko.jokic@florencehc.com