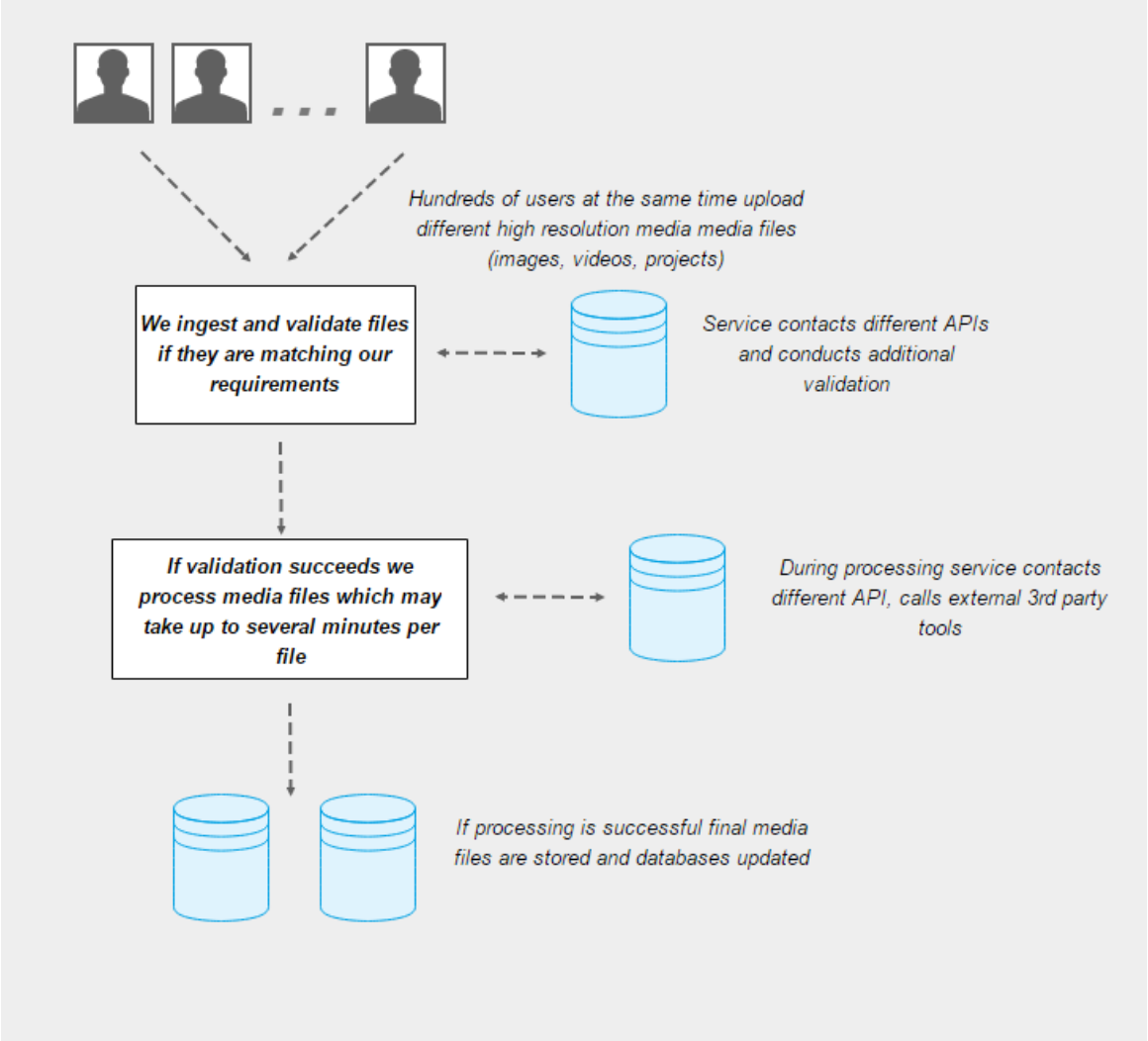
Pond5 Technical Test

Question 1: Architectural solutions design

Please propose a solution on how you would go about uploading, validating and pre-analyzing a huge amount of media files (anything up to and exceeding TBs) using 3rd party services, and then processing them and persisting them into a database.

The below graph illustrates the system you must design.



Additional requirements:

- Make sure that your solution can be scaled up or down depending on the volume of data.

- Solution can handle 3rd party API calls, failure or long response all without blocking the whole service/application.
- There is no single point of failure that can kill the application.
- Feel free to propose any changes/improvements to your design, explaining the advantage of said changes.

Question 2: Item service design

The objective of this code challenge is to develop a backend service or api.

The api responses will use **Json** over **HTTP**.

We would like the service to support CRUD operations (create, read, update, delete) on an item object. The item object will have the following attributes:

```
-id: int
-file_name: string
-media_type: string
-created_at: time
-updated_at: time
```

For example, an example call to the service could be something like:

```
GET /items/1
```

Response:

```
{
    "id": 1,
    "name": "Pond5 digital watermark",
    "media_type": "mov",
    .........
    .........
}
```

Please use Python and make any decisions and assumptions you need or want.

What we need to be able to consider the test:

- Provide the code for the service in a Git repo (Github, Bitbucket, Gitlab is fine)
- Please provide anything we need to understand and run your project easily. The easier to run locally the better.

There is no set time limit to complete the test, but candidates will be evaluated in order of returning the test.

Any questions please contact us.

Good luck.