RoboVoice Technical Challenge

Intro

Your challenge is to build an [Imgur](#) image uploading service that exposes a REST API.

Clients of this service:
- Submit image upload jobs
    - Each upload job is an array of image URLs. The service takes each URL, downloads the content, and uploads it to Imgur.

Optional: The service should be Dockerized.

Requirements

- Implement the API outlined in the next section
- Do not use Imgur's upload by link functionality - the service needs to download the image at the given URL and re-upload to Imgur.
- Use your choice of language/platform
- OAuth client ID and secret can be embedded into the service config or injected at runtime.
- Do not use any external datastore - use in-memory data structures.
- `POST /v1/images/upload`
    - Should return immediately. The uploading happens asynchronously.
    - Should log the reason for failed image uploads.
- Write unit tests to test your functions
- Delete print() & test code that you used for testing
- (Optional) Finally, please add a Dockerfile
- If anything is unclear, you can come up with your own assumptions. Please also state them in the README together with instructions on how to run the service and any documentation that you think is necessary

API

Submit image URLs for upload

Submits a request to upload a set of image URLs to Imgur. The images will be uploaded to the configured Imgur, viewable by anyone with the link.

Request

```
POST /v1/images/upload
```

The request is just a JSON body, no query parameters.

Request body

Attributes:
- urls: An array of URLs to images that will be uploaded. Duplicates should be stripped out.

Example request body:

```
{
    "urls": [
        "https://farm3.staticflickr.com/2879/11234651086_681b3c2c00_b_d.jpg",
        "https://farm4.staticflickr.com/3790/11244125445_3c2f32cd83_k_d.jpg"
    ]
}
```

Response

On success, returns immediately with an appropriate status code with the id of the job.

Response body

Attributes:
- jobId: The id of the upload job that was just submitted.

Example response body:

```
{
    "jobId": "55355b7c-9b86-4a1a-b32e-6cdd6db07183",
}
```