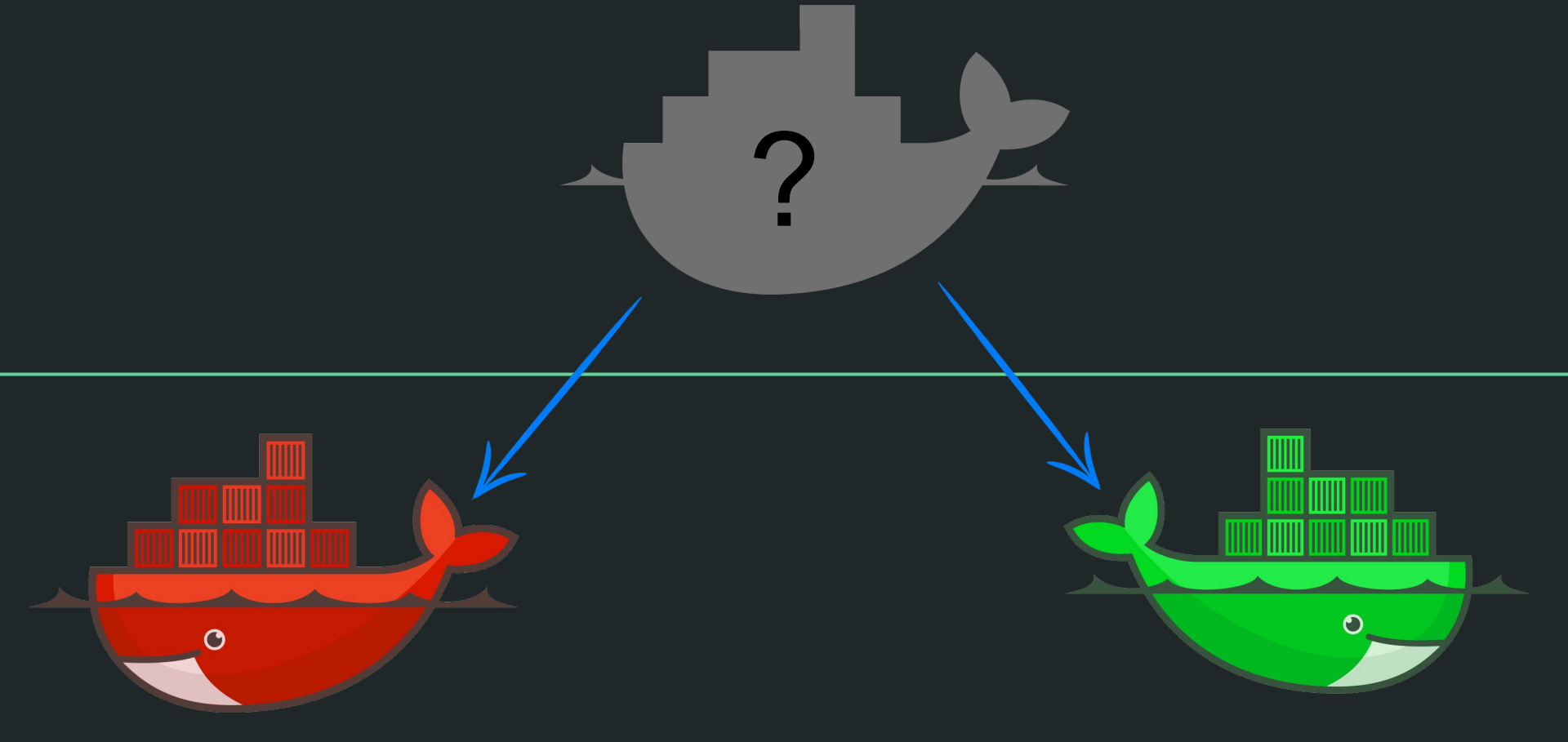
Container Code Classification

Gao, Jeremy, Kostas, Ozan, Rahul

Mentor: Sastry S Duri
(IBM Research)

The Goal: to Classify Safety



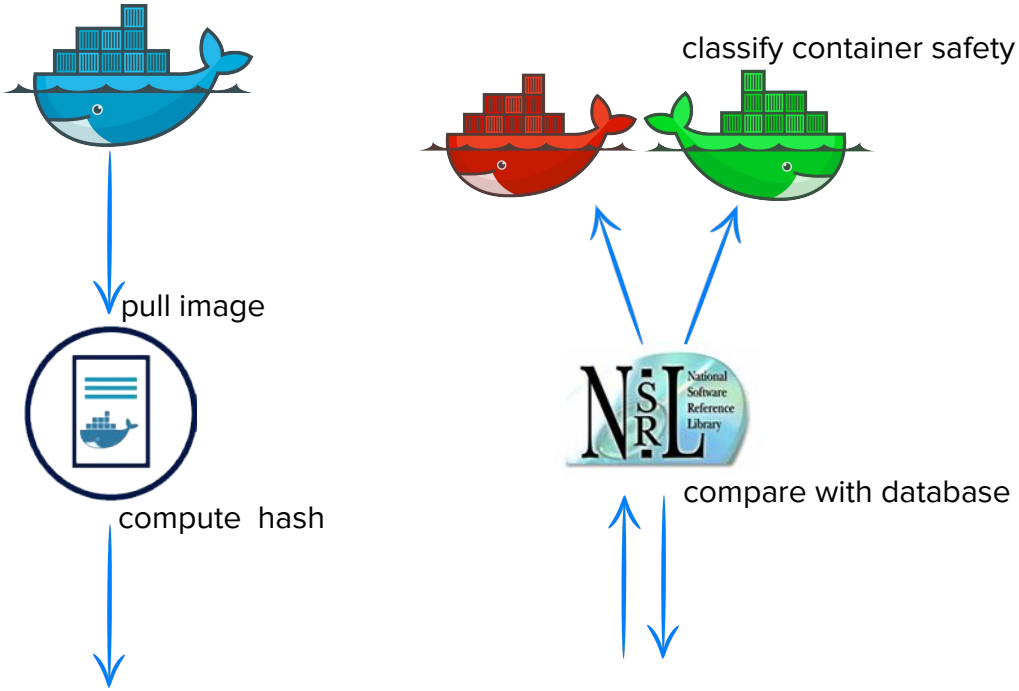
How can we detect suspicious files in a docker container?

Research Project Goal:

- Find a way to reduce the dataset to a number of suspicious files in an image
- We are testing different models and comparing them to find the most accurate prediction of the safety of a container

Procedure:

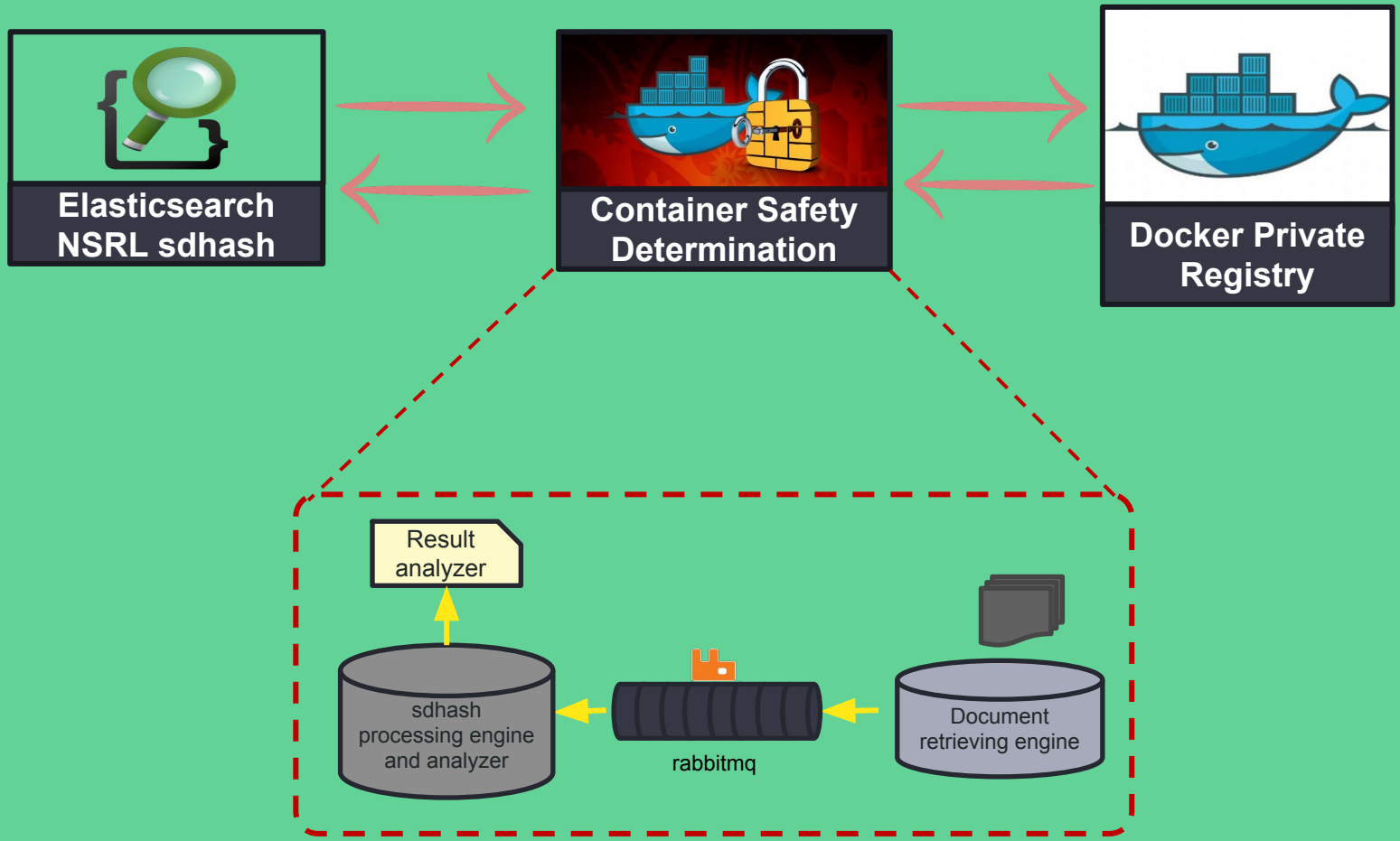
1. Retrieve
2. Hash
3. Compare
4. Classify



FAAAAAICAAARUAAAAAAAQAAAAAACEAAQQCAAAQAAkAAEAQAAJAAAgEAAgIIABIQA
AAAAoEBAgAAABABAAAAAAAAAEAgAAgAAACAIAAAEAAAAAAAABAAACAAAAAAgDAA
AAAAAAAAgAAABAAgAAAACQAAIAAAAAAAAAAAACAAIEAAAAAgwAAAAQAAAAAQAA
QBAEAAKAACAAAAAAgDAAAAAAAAA

Technologies

- SDHash
- Docker
- ElasticSearch
- Rabbitmq


Elasticsearch
NSRL sdhash
Container Safety
Determination
Docker Private
Registry
Result
analyzer
sdhash
processing engine
and analyzer
rabbitmq
Document
retrieving engine

Why sdhash?

crypto-hashes not suited for similarity comparison:

small change in a file will generate completely different crypto-hash.

sdhash: Similarity digest hash

Reduces the file size to 2-3% of the original

sdhash tool compares 2 files and generates a similarity index 0-100

Example: sdhashes for two text files (~14KB) with mostly same content except a few sentences deleted from file2.

file1 hash

```
sdbf:03:9:file1.txt:14030:sha1:256:5:7ff:160:2:77:CIEVAjQGU
EAHAASiSBZowLB3B4FAZDQAADKGoFQNHllAAlkDqmAA8AQOMJjIHQGSTAnA
cmBCtDQDtJIwCoGInUkSDybgowgdEwQHFllBiIBYMyjKFYspoDzsAlxrFEL
iiLogDUQAQOESKiQAIBAAyCgQGaDBKAIO+LgAYE5DA5mQmmZgwAgiBAEmA9
AAMiLBAUOgOQoAiGrIg7oSJtGObOBIQEEJAAwCQ4ll2ENeAUgmAOyU4ICzq
ECVhowoxVUwJ2CAAXBkDqNyiiAoQA/E8FlrYQF4QRAAQQi2lqDEtwSLKEAZ
ITuzD6gooJIr+KD2QSQIBzSFkasBAqQACz4AERqggAAAEIAlAUNoGCIAACT
AAAICCGWHIICwAQkgAkAEBBAQBAFEAgQAgAAAAARABEGAMgBYiAAYAGAQAK
IAKFAQCBCwIgAUCBLDICAGCgIAURAAEAkgAIAEBAAmVxMALBQBAAQQCIigF
CACAAICAAgQEhFREAgggIAAgwEghgABEAgQihSAAFAoLATUAADAASEAgABQ
BIUiABAiQAAEIAIDAAAAIggCJCARABgAJSApCaAwIAIBggwwAAAUUAINAIE
ABKEBkAIAMAECqEMIkNGpAoCFAkBCQAAAFBgAgAAwAB2BAAUEgA6AADIVAI
BEwMAIyEBQQAoAAYAAggeBGiA=
```

file2 hash

```
sdbf:03:9:file2.txt:13607:sha1:256:5:7ff:160:2:71:CIEVAjQGf
OEAGAASiSBZowLB2BYFAZDQAADKGwFQJHllAAlkDqmAA9AQOMJjIHwGSTAn
gcqBCtDQDtJIwCoGInUkSDzaioxgdAwQHFllBiIBYMyjKFYspoDzsAlxrFE
KiiLogDUQAQOECKiQAIBAAyCgQGaDBKAIK+LgCYG5DA5mQimZgwBggBAEuA
9AAMiDBAUOgOQoAiGrIg7pSBtGOaoBIUEEpAAxCA4ll2ENeAEgnAOzU4IKz
qECVhowoxVUwJOCEAXBkDqNyiiAoQA/E8Flj4QF4QRBBQQg2liDEtwSLKEE
ZATuzD6gooJIp+KD2QSSIBzCFOasBAqSACz8AERoggAAAEIAlAUNoGCIAAC
TAAAICCGGHIICwAQkgAkAEBBAQBAFEAgQAgAAAAABABEGAEgBYiAAYAGAQA
KIAKFAQCBCgAAAECBLDICAGCgIAQRAAEAkgAIAEBAAmVRMALBQBAAQQCIig
FCACAAICAAgQEhFREAgggIAAgwEAhgAAEAgQihSAAFAoJATEAADAASEAgAB
QBAUiABAiAAAEIAIDAAAAAggCJCARABgAJSApCaAwIAIBAgwwAAAUUAINAI
EAAKEBkAIAMAECqEMIkNGpAgCFAkBCAAAAFBgAgAAwAB2AAAUEgAyAADIVA
IBEwMAIQEBQQAoAAYAAggaAGiA=
```

sdhash similarity index: 96

Why NSRL?

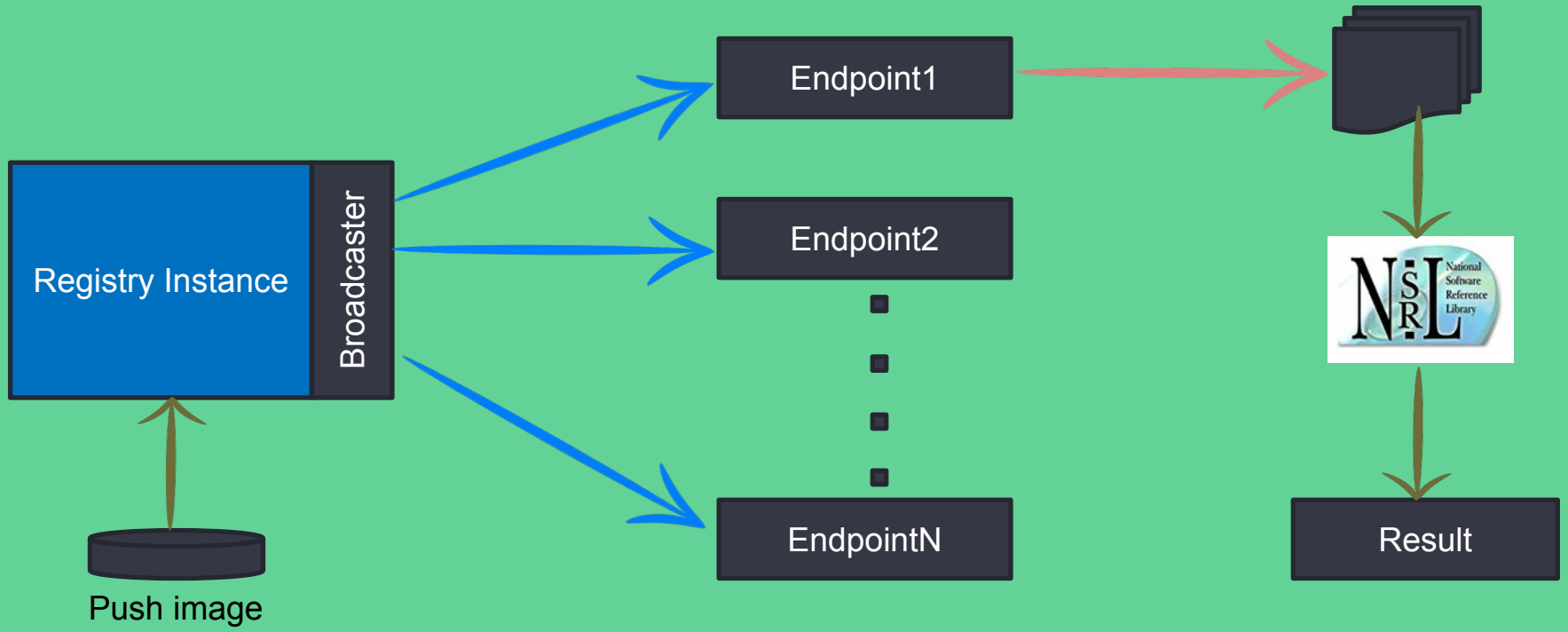
- ➔ Provides SDHash
- ➔ > 40,000,000 hashes
- ➔ Quarterly Release

Goal: utilize the precomputed hashes provided by NSRL

Docker Registry

- Manage docker images
- Multiple storage backends (local, S3, Ceph, ...)
- Supports notifications to update its status to endpoints

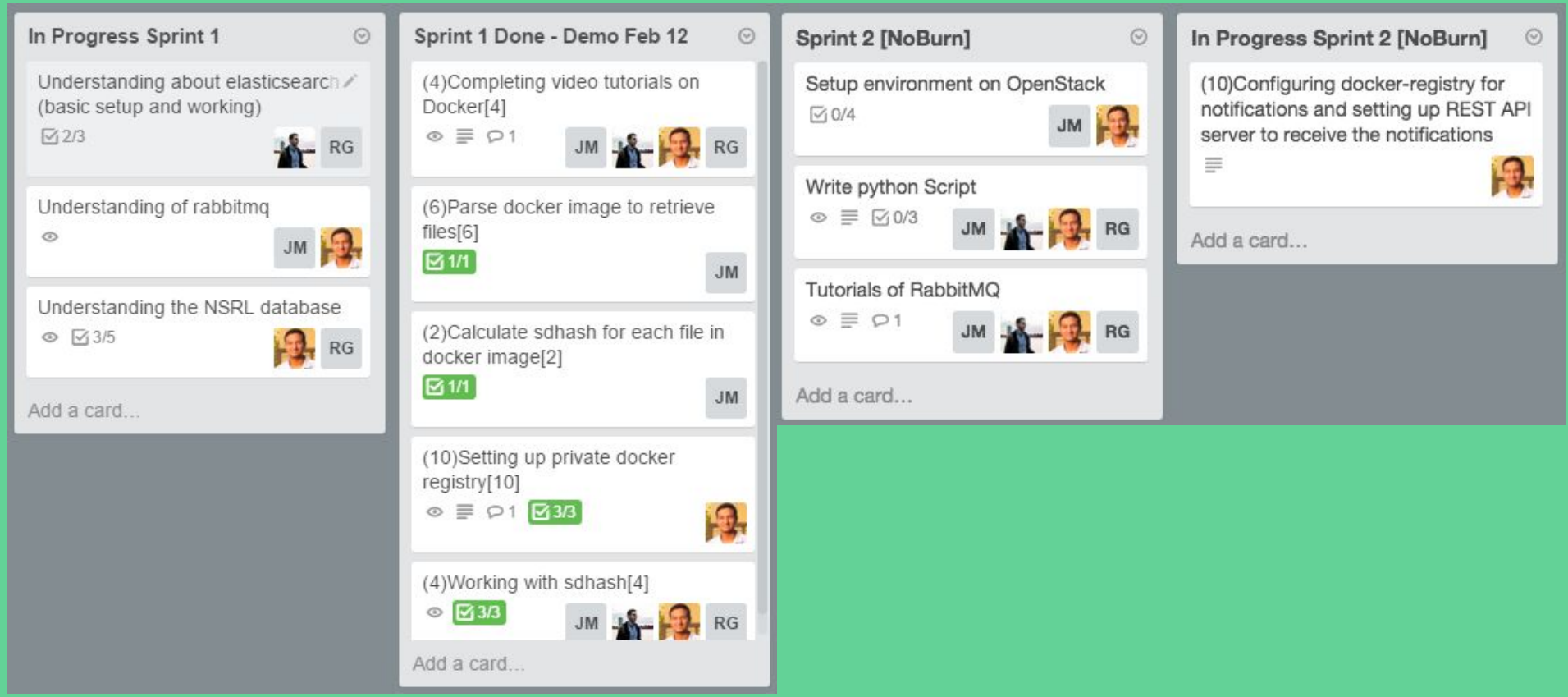
Registry Notifications and sdhash



Sprint 1 tasks

- Private Docker registry
- Script to pull and extract files from Docker registry
- Setup private environment
- Sdhash compilation and installation
- Script to calculate sdhash of files extracted

Tasks done and in progress



Next sprint

- First run of data with elasticsearch
- Come up with the sdhash threshold value ranges
- Implement REST APIs for endpoints
- Integrate image push with hash-calculation

Thank you

&

Q/A