Backend Java / Mobile API Developer:

Create a simple web service application using Spring Boot which contains these two services.

This test intended to demonstrate rather that you could apply several software engineering principles (design pattern, SOLID principle, etc) in spring boot application.

Dog Service:

This API is used for listing all of dog breeds and sub breeds.

- This API should consume the data of dog breeds from https://dog.ceo/dog-api/ as data source

Criteria :

- CRUD: Create Dog Restful service with GET, PUT, POST, and DELETE endpoints.
- You can use any embedded in memory SQL database (ie. H2). Attach the Database DDL script.
- Provide Unit Test / Integration Test
- Supply with note, containing answers of this following questions:
  - Tell us, in which line you use spring boot singleton pattern
- Please use following cases :
  - If the data source endpoint name breeds/list/all make rest API client timeout 5000 ms, if the data source endpoint is breed/*{sub-breed}*/list make timeout 2000 ms (*using standard spring boot bean is plus*)
  - Using lambda and method reference is a plus
  - If the dog breed is *shiba*, then you should take only an odd number of images. (*using design pattern for this case is a plus*)
  - If the dog breed is *sheepdog*, then you should extract the sub breed, become the parent breed (*using design pattern for this case is a plus*), example :

From :	To:
``` "sheepdog": ["english", "shetland"], ```	``` "sheepdog-english": [ ], "sheepdog-shetland": [ ], ```

○ If the dog breed is *terrier*, then you should extract the sub breed, become the parent breed and fetch the images using completable futures

<table>
<tr><td>

From :

```
],
"terrier": [
  "american",
  "australian",
  "bedlington",
  "border",
  "cairn",
  "dandie",
  "fox",
  "irish",
  "kerryblue",
  "lakeland",
  "norfolk",
  "norwich",
  "patterdale",
  "russell",
  "scottish",
  "sealyham",
  "silky",
  "tibetan",
  "toy",
  "welsh",
  "westhighland",
  "wheaten",
  "yorkshire"
],
```

</td><td>

TO :

```
"terrier-american": [

    "https:\/\/images.dog.ceo\/breeds\/terrier-american\/n02093428_10164
    .jpg",
    "https:\/\/images.dog.ceo\/breeds\/terrier-american\/n02093428_10245
    .jpg",
    "https:\/\/images.dog.ceo\/breeds\/terrier-american\/n02093428_10328
    .jpg",
    "https:\/\/images.dog.ceo\/breeds\/terrier-american\/n02093428_10365
    .jpg"
  ],
  "terrier-australian": [
   "https:
    \/\/images.dog.ceo\/breeds\/terrier-australian\/FB_IMG_15966253966
    17.jpg",
   "https:
    \/\/images.dog.ceo\/breeds\/terrier-australian\/n02096294_1111.jpg",
   "https:
    \/\/images.dog.ceo\/breeds\/terrier-australian\/n02096294_1121.jpg",
   "https:
    \/\/images.dog.ceo\/breeds\/terrier-australian\/n02096294_1133.jpg"
  ],
```

</td></tr>
</table>

Notes:
- You can use Gradle or Maven.
- You can use Lombok.
- Documentation would be a plus (ie. Swagger).
- Input validation (ie. Spring Bean Validation / Hibernate Validation) would be a plus.
- Addition of caching layer would be a plus (ie: Using redis or in-memory Cache).
- You can submit the cleaned project using .zip (don't use .rar) via Google Drive or email attachment.
- Complete within 3 days but if you need additional time (ie. because of work), please confirm first.

Jika sudah selesai dapat dikirimkan ke hrd@gli.id dan cc ke pramudita.widiatmoko@gli.id dan clara.yarlin@gli.id. Jika ada pertanyaan dapat ditanyakan via Whatsapp di nomor yang telah diberikan