

1) ข้อมูลกลุ่ม และ งานที่แต่ละคนทำจริง ใน project (เฉพาะ back-end)

2) Application URL: http://???.???

3) Endpoints API Description (example)

**Brand:**

####

**GET** <http://int-project-backend/brand/>

>> Get All brand

####

**PUT** <http://int-project-backend/brand/>

>> Update brand by brand code in receive Json

####

**DELETE** <http://int-project-backend/brand/{{brandCode}}>

>> ลบ Brand ตาม brand code ที่ระบุ

####

**GET** <http://int-project-backend/brand/{{brandCode}}/products>

>> แสดงแบรนด์ทั้งหมด ตามรหัสที่ระบุ

**Images:**

####

**GET** <http://int-project-backend/img/{{filename}}>

**Product:**

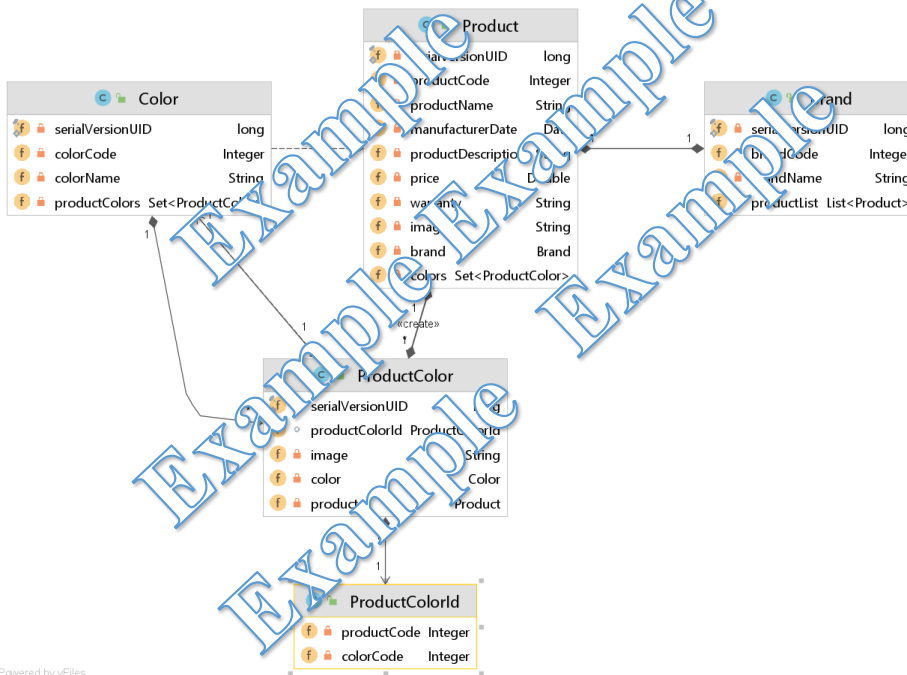
####

**GET** <http://int-project-backend/product/>

####

**POST** <http://int-project-backend/product/>

#### 4) Class Diagram (UML)



#### 5) Application properties file (production)

```
server.port=3000
spring.jpa.hibernate.ddl-auto=none
spring.datasource.url=jdbc:mysql://${MYSQL_HOST:172.18.0.4}:3306/productdb
spring.datasource.username=productapp
spring.datasource.password=zck9!Jz
server.error.include-stacktrace=never

## MULTIPART (MultipartProperties)
# Enable multipart uploads
spring.servlet.multipart.enabled=true
# Threshold after which files are written to disk.
spring.servlet.multipart.file-size-threshold=2KB
# Max file size.
spring.servlet.multipart.max-file-size=32MB
# Max Request Size
spring.servlet.multipart.max-request-size=128MB
## File Storage Properties
# All files uploaded through the REST API will be stored in this directory
integrated.storage.location=./public/product-images
```