

VS-Lab - eShop Report

Gruppe(IZ-Kürzel): krse1019, krse1020, ruf11020, wigo1011

GitHub Repos:

- Legacy-Server: <https://github.com/HuBaX/hska-vis-legacy>
- Product-Service: <https://github.com/HuBaX/ProductService>
- Category-Service: <https://github.com/HuBaX/CategoryService>

Aufgabe 1

1. Aufgabe

InformatikAlle ProdukteProdukt hinzufügenKategorien bearbeitenLogout

Sie sind eingeloggt als admin admin

Suchanfrage

Suchtext:

Preis min.:

Preis max.:

Suchen

Produkte

Nummer	Name	Preis	Kategorie		
1	Messer	10.0	Sonstiges		

InformatikAlle ProdukteProdukt hinzufügenKategorien bearbeitenLogout

Sie sind eingeloggt als admin admin

Produktbeschreibung

Produktname:	Besteckset
Preis:	35.0
Kategorie:	K?chenutensilien
Beschreibung:	4 Messer, 4 L?ffel, 4 Gabeln

[\[Zurück\]](#)

Informatik

Alle Produkte

Add product

Edit categories

Logout

You are logged in as admin admin

Search

Suchtext:

Preis min.:

20

Preis max.:

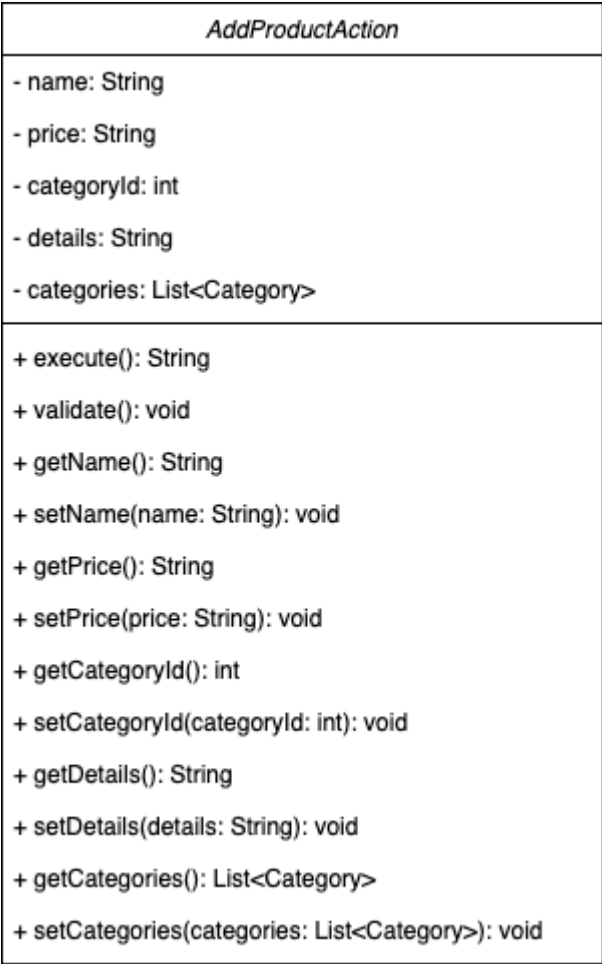
50

Search

Products

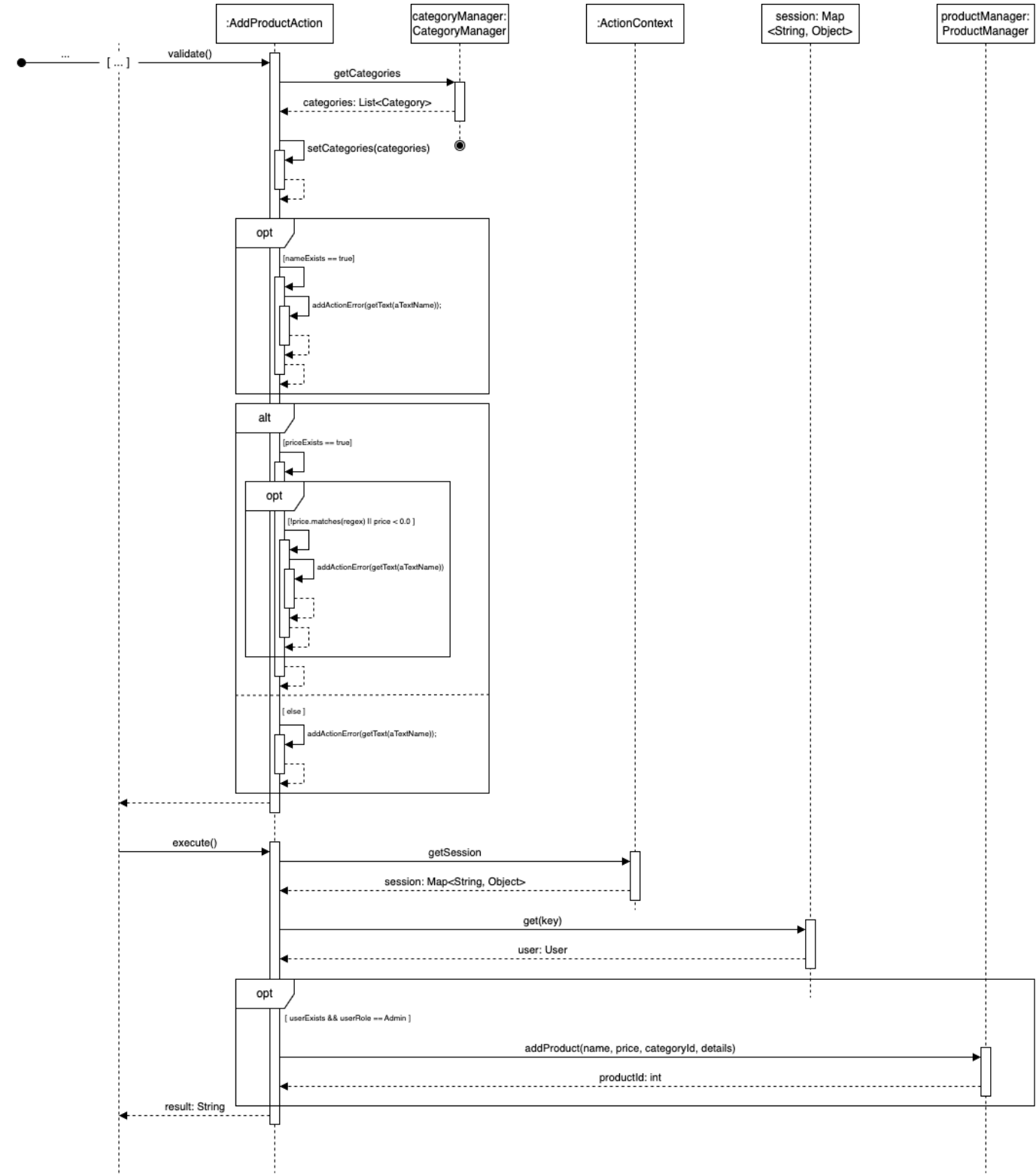
number	Name	Price	Category		
1	Besteckset	35.0	K?chenutensilien		

Class Diagram



2. Aufgabe

Analysieren sie Struktur und Verhalten des eShop --> c - Erstellen sie ein UML Strukturdiagramm (zB Klassendiagramm) und ein UML Verhaltens- diagramm (zB Sequenzdiagramm).



3. Aufgabe

Erstellen sie ein fachliches Makro-modell des eShop mit DDD --> e - Diskutieren sie die Auswahl eines Bounded Context für die Migration als Microservice.

Suche
Produktname Preis

Hinzufügen/Löschen Produkte
Name Preis Kategorie Beschreibung

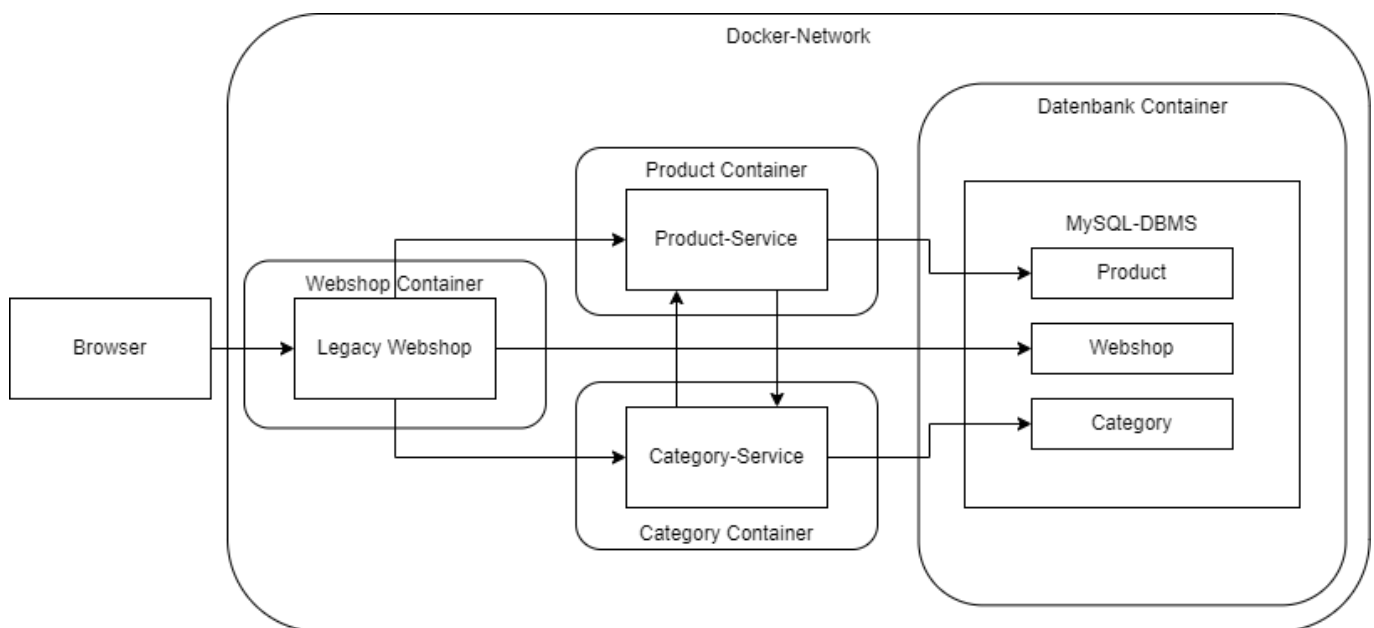
Hinzufügen/Löschen Kategorie
Name

Es ist sinnvoll einen Bounded Context zu migrieren, weil dieser einen in sich geschlossene Domäne repräsentiert und somit keine Überschneidung mit Komponenten aus dem Legacy System hat.

Wenn man sich das Strangler pattern als Vorgehen aussucht, wäre es am sinnvollsten die Komponenten zu migrieren, die am häufigsten von den Entwicklern geändert werden müssen. Komponenten die vielleicht gar nicht mehr geändert werden müssen, können sogar in manchen Szenarien als Legacy System bestehen bleiben. So oder so muss wenn die Migration Schrittweise erfolgt eine Schnittstelle zwischen legacy System und den neuen Microservices geschaffen werden.

Aufgabe 2

Architekturskizze des Webshops in Docker-Containern.



Aufgabe 3

In diesem Abschnitt finden Sie Screenshots zu den verschiedenen Frontends und der Arbeit des Load-Balancers.

Minikube und Docker images

- Ausgabe der Liste aller Docker images auf dem System

```
PS F:\Uni\Master\Semester 3\vsLab\hska-vis-legacy> docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
mavogel/hska-vis-legacywebshop	latest	a1d11d194f9c	19 minutes ago	162MB
mavogel/hska-vis-web-shop-db-image	latest	0b9111d9b0d0	30 minutes ago	360MB
apache	latest	0a3ce96cdc83	31 minutes ago	223MB
category	latest	e1c68b212385	32 minutes ago	1GB
product	latest	65f68a874e49	32 minutes ago	1GB
golang	1.19	aa2a9b347a08	21 hours ago	992MB
ubuntu	20.04	88bd68917189	5 weeks ago	72.8MB
istio/proxyv2	1.17.1	8773d9508106	3 months ago	256MB
istio/pilot	1.17.1	699d9171558c	3 months ago	198MB
quay.io/kiali/kiali	v1.63	361c13ca69bc	3 months ago	106MB
registry.k8s.io/kube-apiserver	v1.26.1	deb04688c4a3	4 months ago	134MB
registry.k8s.io/kube-scheduler	v1.26.1	655493523f60	4 months ago	56.3MB
registry.k8s.io/kube-controller-manager	v1.26.1	e9c08e11b07f	4 months ago	124MB
registry.k8s.io/kube-proxy	v1.26.1	46a6bb3c77ce	4 months ago	65.6MB
registry.k8s.io/etcd	3.5.6-0	fce326961ae2	6 months ago	299MB
registry.k8s.io/pause	3.9	e6f181688397	7 months ago	744kB
grafana/grafana	9.0.1	b6f09c63d6ec	11 months ago	292MB
jaegertracing/all-in-one	1.35	9d3f84f2ca38	11 months ago	59.9MB
registry.k8s.io/coredns/coredns	v1.9.3	5185b96f0bec	12 months ago	48.8MB
prom/prometheus	v2.34.0	e3cf894a63f5	14 months ago	205MB
registry.k8s.io/pause	3.6	6270bb605e12	21 months ago	683kB
gcr.io/k8s-minikube/storage-provisioner	v5	6e38f40d628d	2 years ago	31.5MB
jimmidyson/configmap-reload	v0.5.0	d771cc9785a1	2 years ago	9.99MB
maven	3.5.4-jdk-8-alpine	fb4bb0d89941	4 years ago	119MB
tomcat	8.0-alpine	624fb61775c3	4 years ago	147MB
mysql	5.7.9	ec7e75e5260c	7 years ago	360MB

```
PS F:\Uni\Master\Semester 3\vsLab\hska-vis-legacy>
```

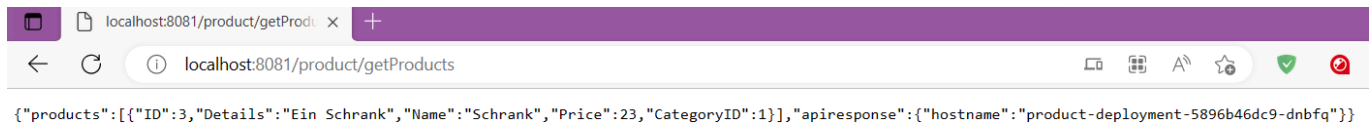
- oAusgabe der Liste aller Kubernetes Pods aus dem Minikube

```
PS F:\Uni\Master\Semester 3\vsLab\hska-vis-legacy> kubectl get pods -A
```

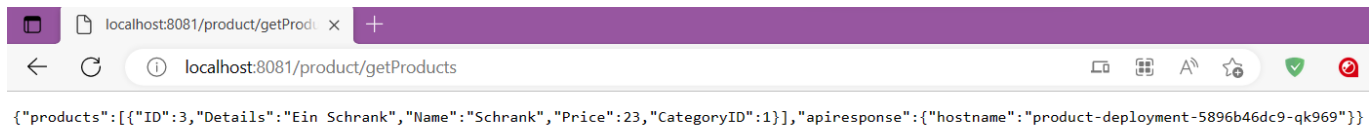
NAMESPACE	NAME	READY	STATUS	RESTARTS	AGE
default	apache-58cf6d5f4b-2jmvh	2/2	Running	2 (24m ago)	30m
default	category-deployment-784c74cbfb-zffv2	2/2	Running	2 (24m ago)	30m
default	database-deployment-598cc6b7fc-qglcj	2/2	Running	2 (24m ago)	30m
default	legacywebshop-deployment-dd66dd6d8-zsw2p	2/2	Running	0	19m
default	product-deployment-774cbd9867-2ldvn	2/2	Running	2 (24m ago)	30m
default	product-deployment-774cbd9867-rfshl	2/2	Running	2 (24m ago)	30m
default	product-deployment-774cbd9867-zng7j	2/2	Running	2 (24m ago)	30m
istio-system	grafana-69f9b6bfdc-w6tkr	1/1	Running	1 (25m ago)	29m
istio-system	istio-ingressgateway-6f86b8c88d-g45xf	1/1	Running	1 (24m ago)	31m
istio-system	istiod-79d65bf5f4-nvqbn	1/1	Running	1 (25m ago)	31m
istio-system	jaeger-cc4688b98-rwvz7	1/1	Running	1 (25m ago)	29m
istio-system	kiali-594965b98c-bm764	1/1	Running	1 (25m ago)	29m
istio-system	prometheus-5f84bbfcfd-tvhv4	2/2	Running	2 (25m ago)	29m
kube-system	coredns-787d4945fb-qcx96	1/1	Running	1 (24m ago)	34m
kube-system	etcd-minikube	1/1	Running	1 (25m ago)	34m
kube-system	kube-apiserver-minikube	1/1	Running	1 (25m ago)	34m
kube-system	kube-controller-manager-minikube	1/1	Running	1 (25m ago)	34m
kube-system	kube-proxy-r8gzf	1/1	Running	1 (25m ago)	34m
kube-system	kube-scheduler-minikube	1/1	Running	1 (25m ago)	34m
kube-system	storage-provisioner	1/1	Running	1 (25m ago)	34m

Load-Balancer

1. Erster ProductRequest



2. Zweiter ProductRequest



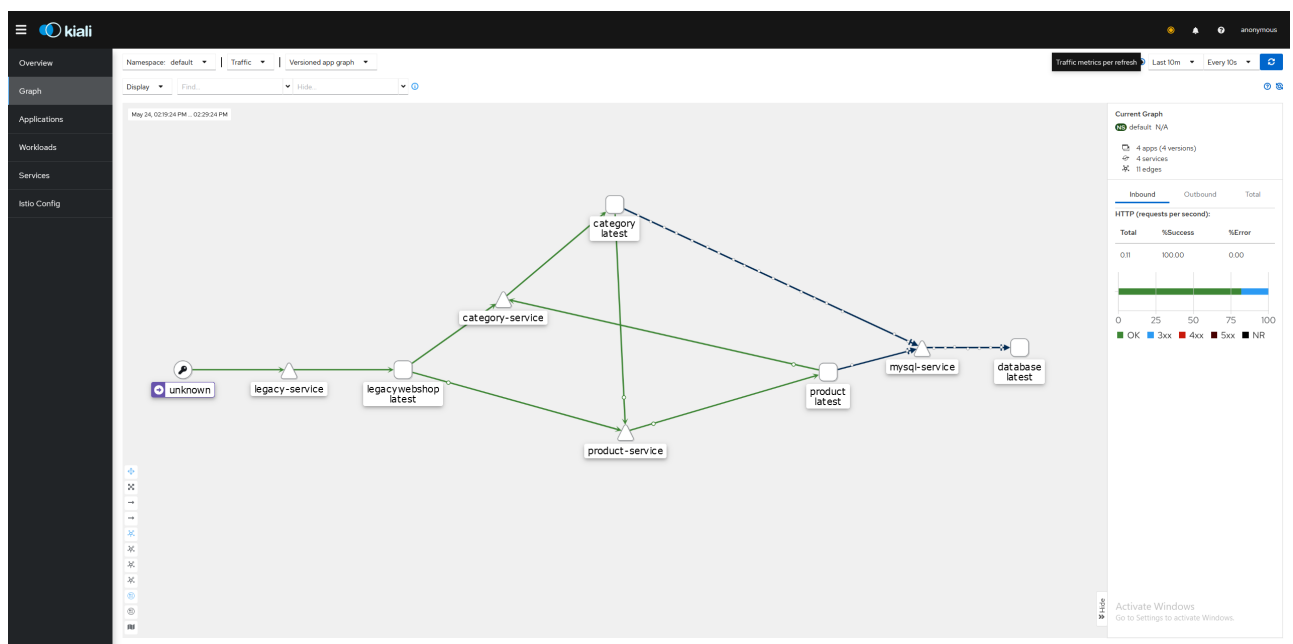
3. Dritter ProductRequest



Alle Hostnamen sind unterschiedlich! -> Der Load-Balancer funktioniert wie erwartet.

Kiali und Grafana Dashboards

- Screenshot von Kiali nach Generierung von Traffic auf dem Webshop



- Screenshot von Grafana(istio-control-plane-dashboard) nach Generierung von Traffic auf dem Webshop



Aufgabe 4

Performancetest Ergebnisse der Responsetime der verschiedenen Requests.

```
Response Time for getCategory: 173
Response Time for getCategory: 15
Response Time for getCategory: 9
Response Time for getCategory: 10
Response Time for addCategory: 14
Response Time for getCategory: 22
Response Time for getCategory: 9
Response Time for getCategory: 8
Response Time for getCategory: 7
Response Time for addCategory: 8
Response Time for getCategory: 7
Response Time for addCategory: 7
Response Time for getCategory: 7
Response Time for getCategory: 7
Response Time for addCategory: 8
Response Time for getCategory: 6
Response Time for getCategory: 8
Response Time for addCategory: 7
Response Time for getCategory: 6
Response Time for getCategory: 7
Response Time for addCategory: 7
Response Time for getCategory: 6
Response Time for getCategory: 8
Response Time for addCategory: 7
Response Time for getCategory: 5
Response Time for getCategory: 7
Response Time for getCategory: 7
Response Time for getCategory: 6
```

```
Response Time for getCategory: 6
Response Time for getProductByName: 13
Response Time for addProduct: 71
list all products!
Response Time for getCategory: 6
Response Time for getProducts: 26
Response Time for getCategories: 8
Response Time for getCategories: 7
Response Time for getCategory: 6
Response Time for getCategory: 6
Response Time for getProductByName: 13
Response Time for addProduct: 53
list all products!
Response Time for getCategory: 7
Response Time for getCategory: 6
Response Time for getProducts: 28
Response Time for deleteProductById: 8
list all products!
Response Time for getCategory: 6
Response Time for getProducts: 13
Response Time for getCategories: 8
Response Time for delCategoryById: 29
Response Time for getCategories: 5
Response Time for delCategoryById: 24
Response Time for getCategories: 6
Response Time for getCategories: 7
list all products!
```



```
Response Time for getProductById: 4
Response Time for getCategory: 6
Response Time for getCategories: 6
Response Time for delCategoryById: 28
Response Time for getCategories: 6
Response Time for getCategories: 7
Response Time for getCategories: 6
Response Time for getCategory: 6
Response Time for getCategory: 5
Response Time for getProductByName: 12
Response Time for addProduct: 37
list all products!
Response Time for getCategory: 6
Response Time for getProducts: 12
Response Time for getCategories: 6
Response Time for getCategories: 6
Response Time for getCategory: 6
Response Time for getCategory: 6
Response Time for getProductByName: 13
Response Time for addProduct: 31
list all products!
Response Time for getCategory: 6
Response Time for getCategory: 6
Response Time for getProducts: 18
```

Avarage Response-Time

Action	Called Action	Avg. Response Time
open add-Product Tab	getCategories()	6ms
add new product	addProduct()	35ms
list all products	getProducts()	45ms
edit categories	getCategories()	5ms
add new category	addCategory()	8ms
delete category	delCategoryById()	14ms
delete product	deleteProductById()	8ms
search product	getProductForSearchValues()	13ms