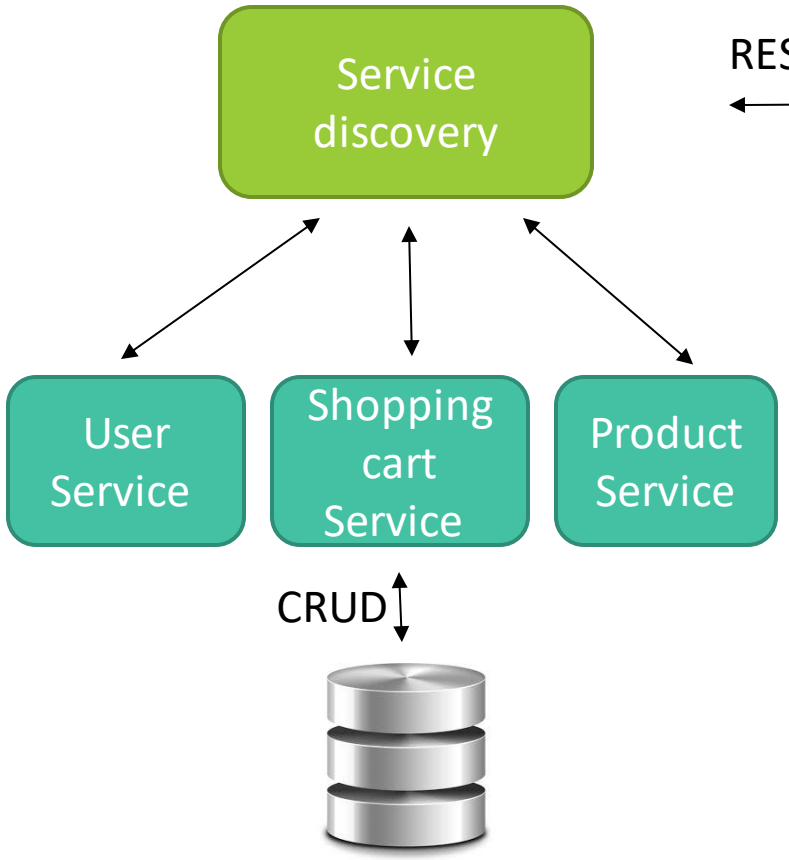
CofeeCat OnlineShop

ADRIAN BEILHARZ

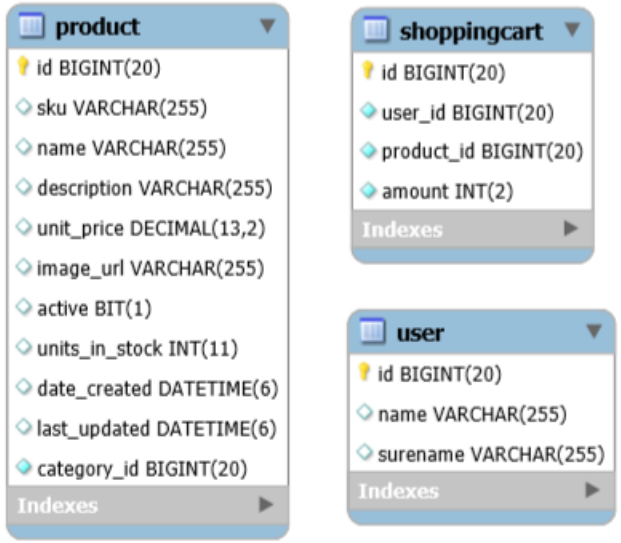
FANNI MAROSI

Project



- Show a list of products
- Add products to shopping cart
- Change amount after adding poduct to the shopping card

Database

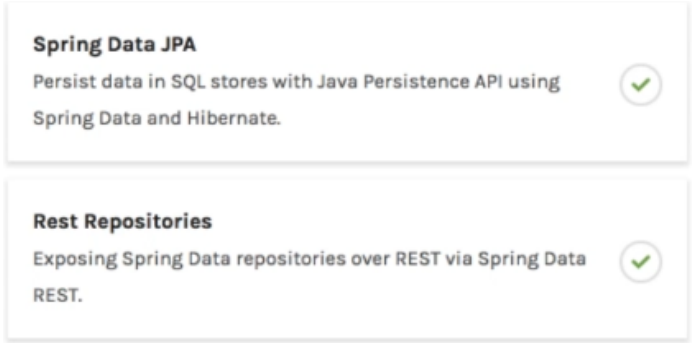


Mysql

Database Shema

Dependencie: MysqlDriver

Repository




```
@CrossOrigin("http://localhost:4200")
public interface ShoppingCartRepository extends CrudRepository<ShoppingCart, Long> {
```

Spring Data REST will scan the project for CRUDRepository (or JPARepository)

Expose REST APIs for each entity type for CRUDRepository

By default, Spring Data REST will create endpoints based on entity type

Service Discovery - Eureka

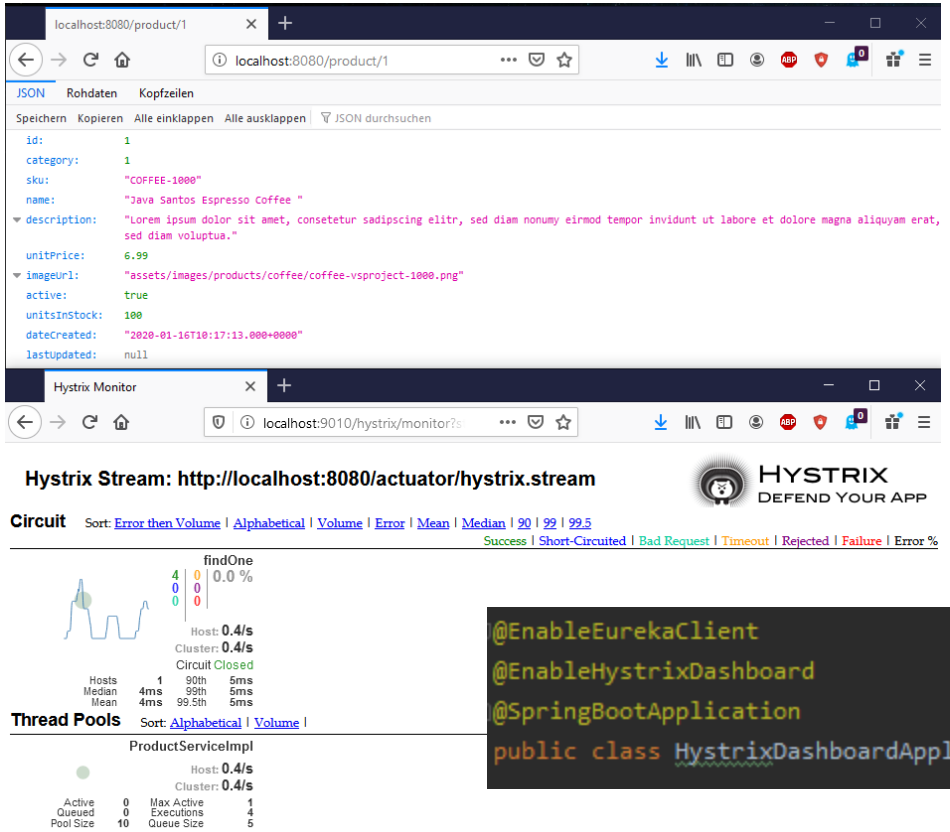
```
@EnableEurekaServer
@SpringBootApplication
public class EurekaServerApplication {
```

AWS Service registry for resilient mid-tier load balancing and failover.

Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
HYSTRIX-DASHBOARD	n/a (1)	(1)	UP (1) - host.docker.internal:Hystrix-Dashboard:9010
PRODUCT-SERVICE	n/a (1)	(1)	UP (1) - host.docker.internal:Product-service:8080
SHOPPINGCART-SERVICE	n/a (1)	(1)	UP (1) - host.docker.internal:ShoppingCart-Service:8082
USER-SERVICE	n/a (1)	(1)	UP (1) - host.docker.internal:User-Service:8083

Hystrix




```
@EnableEurekaClient
@EnableHystrixDashboard
@SpringBootApplication
public class HystrixDashboardApplication {
```

Hystrix is a library developed by Netflix. It is a fault tolerance library and is used as strategy against failures in a service-layer.

Hystrix can be used in situations where the application depends on remote services. In case one or more remote services are down we can handle the situation by using a circuit breaker in the application.

Spring Controller

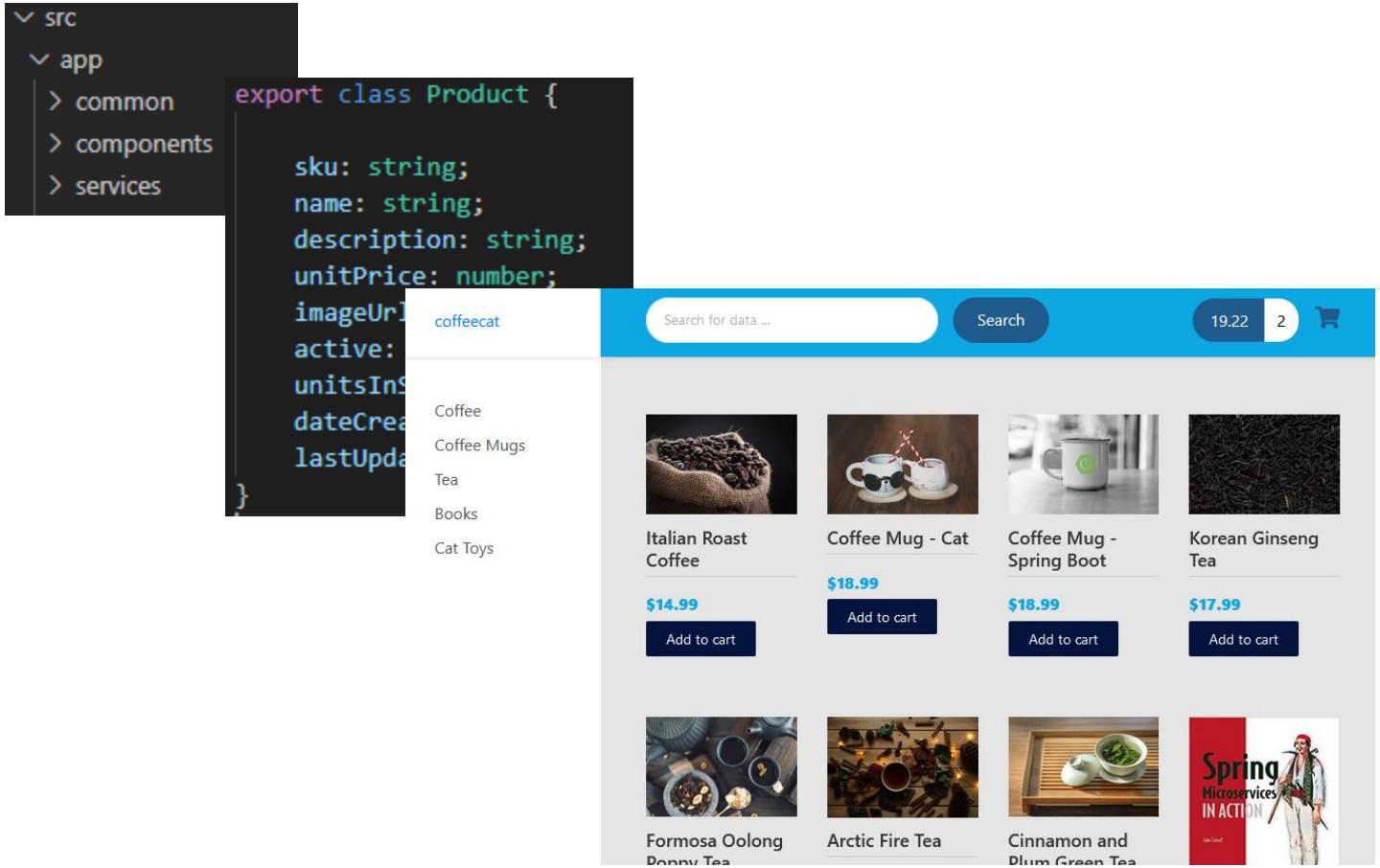
```
@PutMapping
public ResponseEntity<User> updateUser(@RequestBody User user) { return userService.updateUser(user); }

@CrossOrigin("http://localhost:4200")
@PostMapping("/shoppingcart/{userid}/{productid}/{amount}")
public void createShoppingCart(@PathVariable(value = "userid") Long userid, @PathVariable(value = "productid
        userService.addShoppingCartEntry(userid,productid,amount);
}
```

PostMapping

PutMapping

Frontend - Angular



Angular Service to call REST APIs

TypeScript Class for the entities

Angular Components for display

-> CrossOrigin Support : web browsers will not allow (java) script code to call APIs not on same origin(same origin policy)

CofeeCat OnlineShop

THANK YOU FOR YOUR KIND ATTENTION