Latvijas Universitāte

Faculty of Computer Science

Headless e-commerce framework on spring boot and micro-service architecture

Scientific advisor: prof Janis Zuters

Indu Tharanga Samarasinghe Pathirana (is19021)

Abstract

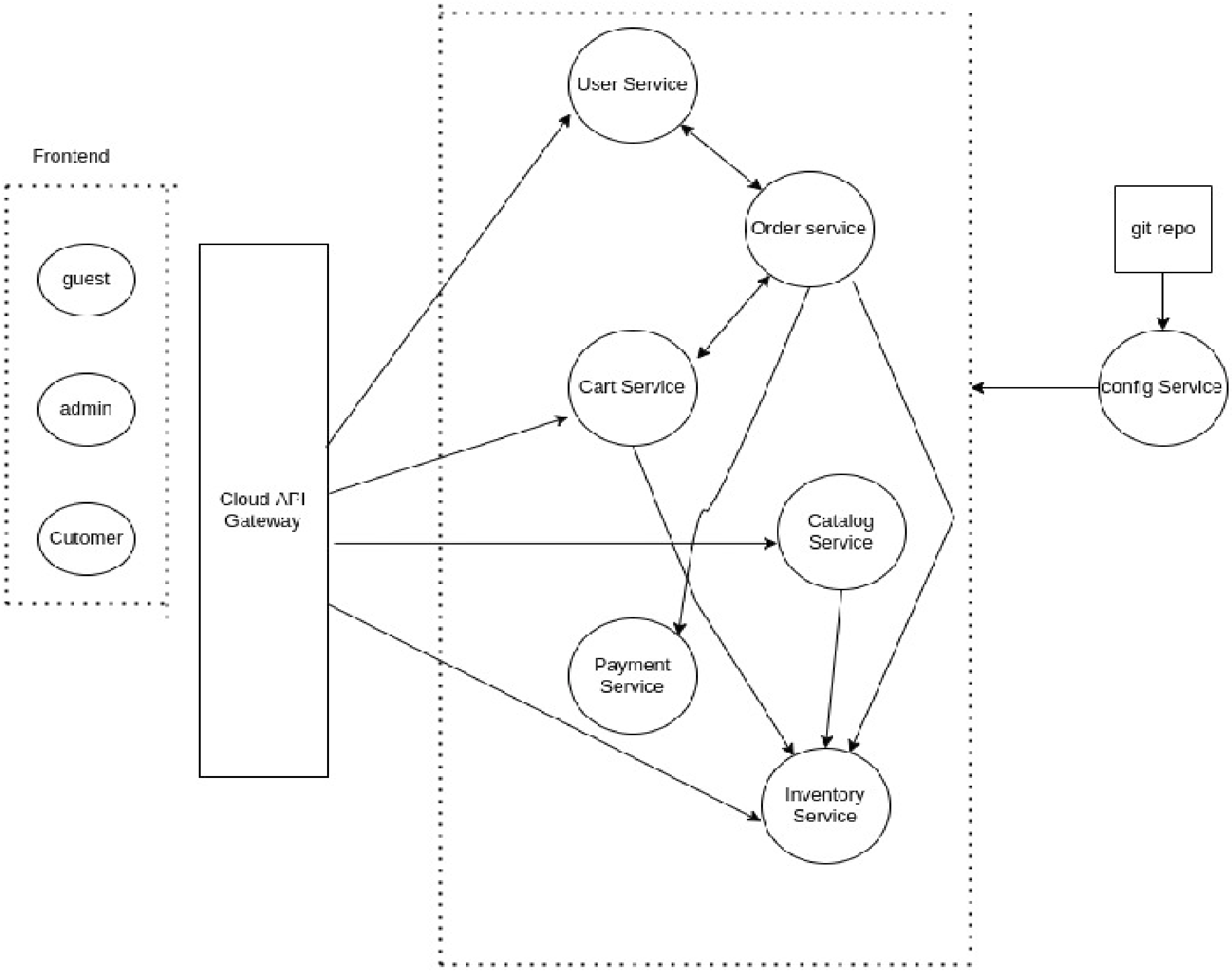
Project work describes an eCommerce framework on Spring boot microservices architecture. Almost all the eCommerce frameworks available in the market are on monolithic architecture. In this work, we break down monolithic architecture components into smaller microservices that can be extended, improve independently, and add new services/features separately on top of that and brings the power of microservices to eCommerce which allows implementing features that are very difficult to achieve with a monolithic architecture.

Table of contents

1.**Introduction**

1.1 Functional requirement

1.1.1 Architecture of the software



1.1.1 Technical requirements

Java 11
Spring boot 2.6.2

2 **User Service**

ID	User group
User-service	Guest, Customer, admin
Name	
User-service	
Description	
Registering, Authenticate , Authorize users	
End points	

Url	Method	Parameters	Description	Return	Authorization
/singup	POST	Email, Name, Password role	Create new customer	USER	Guest
/singin	POST	Email, password	SignIn user	JWT	All
/delete	POST	id	Delete customer	Boolean	Admin,Auth user
/show	GET	id	Get user details	USER	Auth user,Admin
Authorized/{ id}	POST	id	Get User role	user.role	

Output	
Error	

4000 ms of non-response service will call the fall back method and notify the requesting service

3. **Order Service**

ID	User group
Order-Service	Logged in users
Name	
Order-Service	
Description	
Order create, store ,update ,delete	
Endpoints	

Url	Method	Parameters	Description	Return	Authorization
/create	POST	CartObject userID	Create new order	Order	Customer
/delete	POST	oderID	Delete oder	Boolean	Customer, Admin
/show	GET	orderID	Get order details	Order	Customer Admin
/send-payment	POST	OderID	Payfor the order	Boolean	Customer

Output
Error
4000 ms of non-response service will call the fallback method and notify the requesting

service

4 Inventory Service

ID	User group
Inventory-service	Admin ,Interservices
Name	
Inventory-service	
Description	
Create ,keep, and update all inventory details	
End points	

Url	Method	Parameters	Description	Return	Authorized
/add	POST	productID quantity	Add new product	Boolean	Admin
/update	POST	productID quanity	Update inventory	Boolean	
/show	GET	productID	Get user details	StatusObj	
/delete	POST	productID	Delete product record		Admin

Output	
If a user is successfully added to the system, he will	
Error	

4000 ms of non-response service will call the fallback method and notify the requesting service

5. **Catalog Service**

ID	User group
Catalog-service	Guest, Customer, admin
Name	
User-service	
Description	
All Catalog details : Product and Categories	
Endpoints	

Url	Method	Parameters	Description	Return	Authorized
/create-product	POST	productName categoryID	Create new Product	Product	Admin
/create-category	POST	CategorName	Delete new Category	Category	Admin
/get-catelog	GET	/	Get all catelog details	Catelog	All users
/get-product	GET	productID	get product details	Product	All users

Error
4000 ms of non-response service will call the fallback method and notify the requesting service

6. **Cart Service**

ID	User group
Cart-service	Customer,Admin

Name

Cart-service

Description

Create,and maintain user cart and sent to Oder service

Endpoints

Url	Method	Parameters	Description	Return	Authorized
/create	POST	userID	Create new cart	Cart	Customer
/delete	POST	UserID	Delete Cart	Boolean	
/show	GET	UserID	Get user Cart	Cart	Customer,admin
/update	POST	UserID, CartInfor	update User Cart	Cart	Customer,admin
/sent-oder	POST	True	Sendt to Oder Service	Cart	Customer

Error

4000 ms of non-response service will call the fallback method and notify the requesting service

7. **Payment Service**

ID	User group
Payment-service	Customer,admin
Name	
Payment-service	
Description	
Making payments and keep records of all payments	
Endpoints	

Url	Method	Parameters	Description	Return
/pay	POST	OrderObject UserID	Make Payment	Boolen
/get-payment	GET	paymentID UserID	Get payment details	Payment object

Error	
4000 ms of non-response service will call the fallback method and notify the requesting service	

8. **Cloud gateway Service**

ID	User group
Cloud-gateway Service	All users
Name	
Cloud-gateway Service	
Description	
Api gateway to the application from the outside	
Endpoints	
Error	
4000 ms of non-response service will call the fallback method and notify the requesting service	

9. **Config Server Service**

ID	User group
config-server	Guest, Customer,admin
Name	
User-service	
Description	
Registering, Authenticate ,Authorize users	
End points	
Output	
Error	
4000 ms of non-response service will call the fall back method and notify the requesting service	

10. **Hystrix dashboard Service**

ID	User group
Hystrix dashboard Service	,admin
Name	
Hystrix dashboard Service	
Description	
End points	
Output	
Error	
4000 ms of non-response service will call the fallback method and notify the requesting service	

14. Database Design

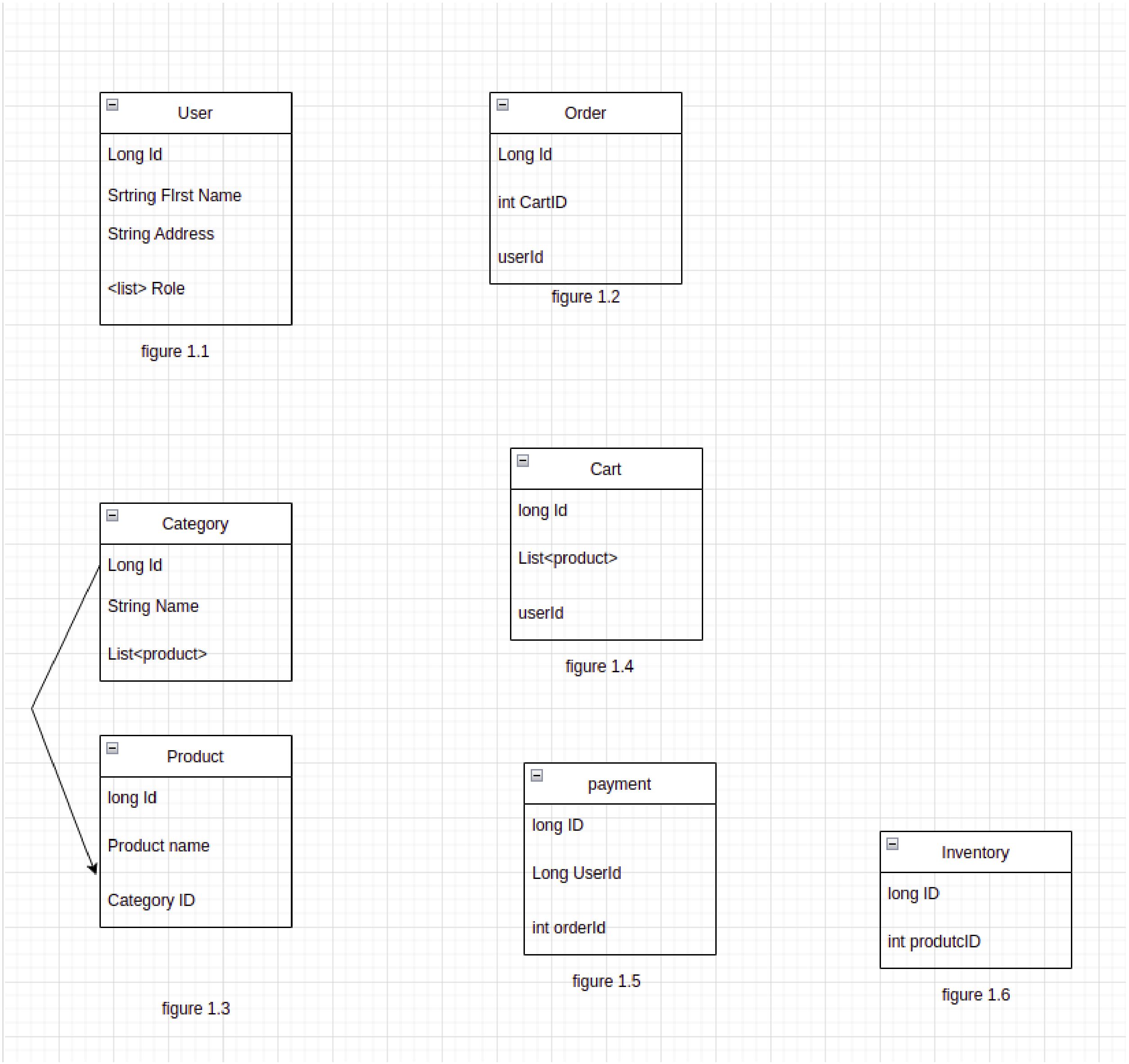


figure 1.1

figure 1.2

figure 1.4

figure 1.3

figure 1.5

figure 1.6

Figure 1

Figure 1.1: User Service database design
Figure 1.2 Order Service database design
Figure 1.3 Catalog Service database design
Figure 1.4 Cart Service database design

Figure 1.5 Payment Service database design
Figure 1.6 Inventory Service database design

12. Security

Users are authenticated by JWT tokens and authorized by the user roles.

13. Future development

Any form of future functionality can be added independently on top of this. For example, multiple complex inventory systems can be added or machine learning services to monitor customer behavior added separately.