E-commerce backend system  
  
Backend : Node JS   
Service Registry : Consul  
Service Discovery : Consul  
Container Orchestration : Docker compose  
Messaging : RabbitMQ  
  
Architecture flow:  
  
A diagram of a service

AI-generated content may be incorrect.  
  
Overview:   
  
This e commerce application is build using Node JS and Express framework .  
  
All the microservice are registered via consul. Consul acts as a service registry, service discovery, monitoring tool.  
  
Product service has a load balancer since product service will be used widely by customer and during peak hours , traffic will increase so load balancer will be helpful in distributing traffic across multiple instances .  
  
Cart and Inventory service does inter service communication using synchronous HTTP protocol using REST API.   
  
Order and Notification service does inter service communication using asynchronous messaging protocol using RabbitMQ .  
  
Project start instructions:  
  
Run command in terminal: docker-compose up --scale product-service=3 --build

Consul Dashboard : http://localhost:8500/

Postman Collection : In project root folder – E-commerce.postman\_collection.json   
  
Services and its docker images:   
  
API Gateway : vigneshactech2015/api-gateway  
Auth service : vigneshactech2015/auth-service  
Product service : vigneshactech2015/product-service  
Cart service : vigneshactech2015/cart-service  
Order service : vigneshactech2015/order-service

Notification service : vigneshactech2015/notification-service

Inventory service : vigneshactech2015/inventory-service