Link: [Spring Cloud Gateway](https://cloud.spring.io/spring-cloud-gateway/reference/html/)

- xây dựng trên nền tảng **pring-webFlux**

- Là cánh cổng(gateway) từ client(mgười dùng) để dẫn vào các **service con(Reserved Proxy)** đang chạy, tránh việc client sd trực tiếp các service con, **gateway** sử dụng **load balancer** của **registry-server** để cân bằng tải giữa các **service con.**

- là project đặc biệt, không thể sd thư viện gateway cùng với thư viện spring-web-starter, cho nên cần loại bỏ **spring-web-starter.**

- cung cấp các thong tin vè  **bảo mật, đo lường, quản lý, khả năng phục hồi khi có service con bị down.**

**Sơ đồ:**

Client <-> gateway mapping <-> gateway web handler <->filter1, filter2 <-> **services**

**Feature:**

Có thể match theo bát kỳ thuộc tính từ link request gửi lên, **dùng predicate để đặt điều kiện cho link, dùng filter để cắt bỏ/lọc đường dẫn trước khi chuyển tới các service con**

**Deps:**

<properties>  
 <java.version>1.8</java.version>  
 <cloud-version>3.1.1</cloud-version>  
</properties>

# sử dụng để lấy config tại server config tập trung

<dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-config-client</artifactId>  
 <version>${cloud-version}</version>  
</dependency>  
  
<!-- https://mvnrepository.com/artifact/org.springframework.cloud/spring-cloud-starter-netflix-eureka-server -->

# theo dõi bởi registry-server  
<dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-starter-netflix-eureka-client</artifactId>  
 <version>${cloud-version}</version>  
</dependency>  
  
<!-- https://mvnrepository.com/artifact/org.springframework.cloud/spring-cloud-starter-gateway -->

# Khiến project trở thành 1 gateway-service  
<dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-starter-gateway</artifactId>  
 <version>${cloud-version}</version>  
</dependency>

**Application.yml:**

spring:  
 application:  
 name: gateway-service  
*# where to load config file* config:  
 import: configserver:http://localhost:8888  
*# do not run when can not load configs* cloud:  
 config:  
 fail-fast: true

**Tại** shared/application.yml của server **config tập trung:**

server:  
 port: 8080  
 servlet:  
 context-path: /  
*# Sử dụng tên ứng dụng XXX-SERVICE (IN HOA) (cột application trong đường dẫn của register-server) để làm đường dẫn truy cập tới các APi  
# nhưng khá bất tiện khi tên của các server đăng kí trong register-server dài.  
#spring:  
# cloud:  
# gateway:  
# discovery:  
# locator:  
# enabled: true*

*# Nếu enable bên trên, thì disable phần này.*spring:  
 cloud:  
 gateway:  
 routes:  
 *# Id định danh của route - optional* - id: user\_route  
 *# SD Tên service trong load-balance của registry-server để điều hướng tới service account-service* uri: lb://account-service  
 *# Path client sẽ truy cập để được điều hướng đến account-service* predicates:  
 - Path=/user/\*\*  
 *# Bởi vì predicates có path là /user/\*\* nhưng trong project account-service lại k có /user/ tại @Requestmapping hoặc servlet-context-path  
 # Cho nên cần StripPrefix=1 để loại bỏ 1 đoạn /XXX/ đầu để đúng với mapping của account-service* filters:  
 - StripPrefix=1  
*# strip: lột bỏ, loại bỏ* - id: statistic\_route  
 uri: lb://statistic-service  
 predicates:  
 - Path=/statistic/\*\*  
 filters:  
 - StripPrefix=1  
 - id: email\_route  
 uri: lb://email-service  
 predicates:  
 - Path=/notify/\*\*  
 filters:  
 - StripPrefix=1

Với port: 8080, Từ giờ, ta có thể truy cập vào bất kì service nào chỉ với đường link**: localhost:8080/path/\*\***

Vd với config Tại **shared/application.yml bên trên,**  ta có 2 service đã được config là account-service và statistic-service tương ứng có path lần lượt là: user và statistic.

Vd: **localhost:8080/user/api/account**: gọi tớiaccount-service, api lấy các tài khoản

**Localhost:8080/statistic/api/statistic:**  gợi tới statistic service lấy các log đã lưu.

**finally:** thêm @EnableDiscoveryClient vào class **Application.class** để đăng ký vào registry-server

RUN.